昭和62年8月1日発行（毎月1回1日発行）第6巻8号通巻64号　昭和58年11月2日第三種郵便物認可

ISSN 0910-7614

パーソナルコンピュータ・マガジン

Oh!MZ

MZシリーズ
X1/turbo
X68000
&ポケコン

特集
迷宮の日本語処理環境
日本語ワードプロセッサ試用レポート
X68000/MZ-2861/書院シリーズ

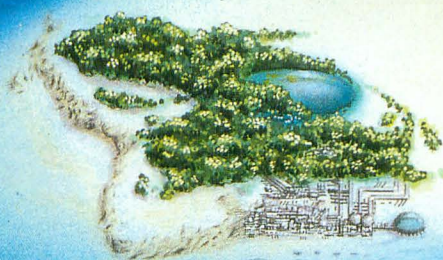
リレー連載 プログラミング実況中継
愛はBASICを育てる

最終回 試験に出るX1
通信プログラムである

新連載 X68000BASIC入門
めぐりあいX-BASIC

X68000あなたの知らない世界
スイッチ/スプライト関数
X-BASIC外部関数の作り方

X1/X1turboパズルゲーム
STAR PANIC

MZ-2500 ワープロプログラム
Superものかきくん

THE SOFTOUCH SPECIAL
Z'sSTAFF PRO 68Kの世界

S-OS全機種共通システム
漢字出力パッケージ
FM-7/77/AV S-OS"SWORD"発表

8
AUG.1987
定価480円

書院ワープロ機能とMS-DOS™ V3.1を標準装備して新しい実務環境を実現。

オフィスでつかうなら文書づくりにこだわったパソコンがいい。そこで専用ワープロの能力を16ビットパソコンに搭載した「MZ-2861」。データ処理、文書処理の両面からあなたのビジネスを強力にサポートします。

ワードプロセッサ"書院"の充実した日本語処理機能を採用(2800モード)

日本語ワードプロセッサ「書院28」搭載：JIS第1/第2水準漢字ROMはもちろん、人名・地名を含む約10万語(内9万語はROM)の辞書を内蔵。企業、団体名をはじめとする固有名詞など、これまで面倒だった宛名書きもスムースに、さまざまなビジネス文書が手軽に作成できます。かな漢字変換も、複数の文節をまとめて効率的に変換できる連文節変換を採用。オペレーションも驚くほどスピーディに。また半角文字、拡大文字、多彩なかざり文字印字など豊かな表現力、そして高度な編集機能を装備しました。パーソナルからビジネスまで幅広い機能をもつ専用ワープロ「書院シリーズ」(3.5″FD内蔵モデル)[※1]の文書も利用できます。

強力な日本語入力(フロントエンド)機能：ビジネスワープロとMS-DOSが融合したフレンドリーな実務環境を実現。本機で作成したワープロ文書や「書院」の文書ファイルとMS-DOSアプリケーションとの間でデータの相互利用はもちろん、MS-DOS上のアプリケーションで日本語入力フロントエンドが利用でき、人名・地名を含めた連文節変換によるスピーディな入力が可能です。

多彩なビジネスアプリケーションに対応する高水準のハードウェア環境(2800モード)

CPUに80286(8MHz)を搭載し高速処理を実現。 別売の数値演算プロセッサのサポートで、さらに処理速度の向上がはかれます。またメモリもメインRAM768Kバイト、ビデオRAM512Kバイトを標準装備。さらに別売の1MバイトRAMボード及び1Mバイト増設RAMにより、最大6MバイトのRAMディスクを本体内に内蔵可能。ハードな実務に対応する大容量メモリを実現しました。グラフィックスも640×400ドットモードで65,536色同時表示を実現、多彩なビジネスグラフや高度なC.G.に対応します。

■スーパーMZのソフトウェアが使える2500モードを装備。MZ-2500シリーズの豊富なアプリケーションを利用できます。[※2]

■フレンドリーな日本語入力のための多機能キーボードを装備。「変換」、「無変換」キーはもちろん、「前候補」、「取消」キーも採用。また、多目的に使える特殊機能操作用のスペシャルファンクションキーも装備しました。日本語ワードプロセッサ「書院28」に対応したキーボードです。

高機能バンドルソフトウェア

日本語ワードプロセッサ「書院28」	：連文節変換をサポートする強力な日本語入力機能、多彩な編集機能、専用ワープロ「書院」(WD-5000シリーズ)のワードプロセッシング能力[※3]を装備。MS-DOSアプリケーションのファイルも利用できます。
MS-DOS™ V3.1	：SET UP、KEY、MKCNF、ファイルコンバート等、多彩なユーティリティを装備した使いやすいディスクオペレーティングシステム。快適な環境でコンピュータが操作できます。
BASIC-M28	：MS-DOS上で動作するBASICです。MZ-2500シリーズのBASIC-M25をベースに、互換性を最大限に保ちながら強力な命令、仕様を追加。MS-DOSのファイル管理を使用しています。

※MS-DOSは米国マイクロソフト社の商標です。

※1 WD-5000D 5000S、5010D 5010Sはメディアをそのまま利用可能。WD-530 535、600 605、610 615、630 631 635は内蔵のデータ変換ソフトにより利用可能。(注)書院カルク、グラフ、図形は利用できません。

※2 ボイスレコーダ、2000 80Bモード、MZ-1E26、MZ-1M08及びRS-232C(Bチャンネル)は使用できません。

※3 書院カルク、グラフ、図形、その他一部の機能で使用できないものもあります。

「書院28」の高度な文書処理機能

- 日本語変換：連文節変換(短縮変換、学習機能、複合語処理、文法解析、接頭、接尾語、連濁処理等)
- 文字サイズ：半角／全角、4倍角、N倍角(24倍角まで)、上つき／下つき、ルビ、ロゴ
- 編集機能：センタリング、右づめ／左づめ、インデント、均等割り付け、桁揃え、枠あけ、複写、移動、置換、検索、禁則、再変換、レイアウト表示、レイアウト表示入力、切り貼り
- 入力方式：記号入力、ローマ字入力、音訓入力、外字入力、区点／JIS入力、シフトJIS入力、部首入力、一括入力、穴うめ、欧文モード
- 文字装飾：罫線、アンダーライン、網掛け、強調文安、斜体、回転、白抜き／黒べた／立体
- 印字機能：簡易印字、差込印字、袋とじ印字、行間指定、字間指定
- その他：ユーザ辞書、分野別辞書、一時登録、外字、時刻機能(時刻・日付)、演算、手続き(プロセス)、縦書き表示、白画面、黒画面、はがき印字、ラベル印字

8ビットMZシリーズ

これから始めたい人に……
ちょっとぜい沢な入門機。

MZ-2520 標準価格159,800円

※14型カラーディスプレイMZ-1D26標準価格89,800円は別売。

さらにグレードを求める人に……
可能性をひろげる高機能。

MZ-2531 標準価格199,800円

※14型カラーディスプレイMZ-1D22標準価格108,000円、モデムホンMZ-1X19は別売。
また装着されているカセットテープは撮影用で、本体の付属品・市販品ではありません。

MZ-2531▲

MZ-2520▶

Oh!MZ

AUGUST 1987

8

表紙絵：Nagasawa Shigeru

CONTENTS

特集

THE SOFTOUCH

ミニ書院WD-260F

Superものかきくん

らくらくSYMBOL

Z's STAFF PRO68K

シリーズ全機種共通システム

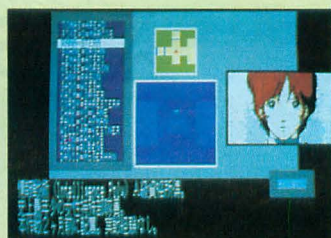
講座/紹介/ゲーム/ビジネス/システム

80386 (開発:インテル 1985年)
8086, 80286と上位互換性のある32ビットマイクロプロセッサ。4段階プロテクトレベルのセグメントとページング方式をサポートしたMMUを内蔵。6段パイプライン,内/外部キャッシュなど高速処理を指向している。CHMOS。内部処理単位32ビット。ピン数132(アドレスバス32ビット,データバス32ビット)。論理アドレス空間64Tバイト/タスク,物理アドレス空間4Gバイト。基本命令数123。最大クロック20MHz。

■広告目次

《スタッフ》
●編集長/前田 徹 ●編集/土平章博 永野 仁 植木章夫 石塚康世 三上之彦 ●協力/有田隆也 高野庸一 西畑文広 Itti Rittaporn 中森 章 清水和人 後藤貴行 林 一樹 近藤弘幸 浅野恵造 山村 一 茗原秀幸 小森 隆 井本 泰 山田伸一郎 堀内保秀 吉田幸一 瀧山 孝 藤原和典 ●カメラ/杉山和美 ●イラスト/永沢しげる 山田晴久 小栗由香 ●アートディレクター/中島真子 ●レイアウト/CANART 元木昌子 渡部善光 ●校正/手塚喜美子 千野延明

STAR PANIC

九玉伝

カーマインXI

スペースハリアー

SHARP

●本体+キーボードCZ-600CE 標準価格369,000円●15型カラーディスプレイテレビ CZ-600DE 標準価格129,800円●チルトスタンドCZ-6ST1E 標準価格5,800円

パーソナルワークステーション

X68000 EXE（エグゼ）スクール開催

●参加者全員にX68000オリジナルテレホンカードを進呈●受講料500円

発売以来、パソコンフリークの話題を独占中のパーソナルワークステーション X68000。この高性能マシンに触れるチャンス到来。未体験者の方を対象に、全国主要地区でエグゼスクールを開催します。実際に手に触れて、可能性にあふれた魅力を体感してください。

詳細はお近くの特約店へ。

夢を超えたパーソナルワークステーション。

個人のツールとしての限界を追求したスペックを搭載。金、銀、そして金属の質感までも再現し得るグラフィックスをはじめとした異次元のアビリティが、まさにこのマシンの象徴として光彩を放っています。本来的に人のもつ創造性に応え得る、16ビットの必然。アーティスティックユースやラボラトリーユースに、この最先端の能力を自在に使いこなしてください。アドバンストユーザーの夢が、もっと未来まで包含されて、いま手の届くところに降りてきました。

実装密度を極限まで追求したフォルムー新のマンハッタンシェイプ

単に、スタイリッシュにフォルムー新、といってしまえば簡単ですが、ここにはそうした言葉ではいい尽くせない、チップ技術をも含めた集積技術、実装技術の確かな裏付けがあります。初めての2万ゲートLSI、ハイスピードICをはじめ10に及ぶカスタムICを開発搭載、本来デスクサイドであるべきカタチをデスクトップにまで凝縮しました。知的な、ハイレベルなユースにふさわしいセンシブルなデザインです。

広くリニアなアドレス空間プロセッサの未来を先取りした68000

メモリ空間の制約にしばられていたグラフィック処理にも新たな次元をひらきます。8ビットの延長上の16ビットではなく、その処理能力に明らかに桁の違うプロセッサ。アドバンストユーザーのクリエイティビティに応える高度なシステム環境をサポートします。クロックはハイスピード10MHz。現時点でのハードの在り方へのひとつの解答として、私たちは68000の良心を選びました。

2Mバイトの大容量メモリ、先駆の独立3画面設計

メインメモリは標準で1Mバイト、さらに内蔵で1Mバイト拡張でき、最大12Mバイトまで拡張可能な大容量設計。また68000のもつ広大なアドレス空間を活かして、テキスト(512Kバイト)、グラフィック(512Kバイト)、スプライト(32Kバイト)の3画面を独立構造として装備した独自のメモリアーキテクチャです。

ビジュアルコントロールで思いどうりに、フレンドリーOS、Human 68k搭載

独自のハードウェアには独自のオペレーティングシステムが必要です。というよりこのX68000に限っては、そうせざる得ない特殊なハード環境が存在します。本機に搭載された独自のOSは、このマシンだけがもつ機能をすべてサポートすることはもちろん、日本語化、ユーザーフレンドリー化への解答をも示す全く新しいOSに仕上がっています。システムの起動後のジョブ選択から操作まで、ほとんどの処理をアイコンで表示し、マウスで選ぶビジュアルシェルによるオペレーティング。極論すれば、コマンドを知らなくてもコンピュータが操作できる。それほどまでのフレンドリネスを追求しました。

連文節変換も、マルチフォントも、日本人にふさわしい強力日本語処理

JIS第1/第2水準漢字ROMの搭載はもちろん、約60,000語に及ぶ強力な辞書を装備。ここでも第2水準漢字がサポートされており、人名・地名をはじめ漢字でなければ表現しにくい熟語などもスムースに表示できます。またOS上のかな漢字変換ソフトウェアとして日本語入力フロントプロセッサを採用。2文節最長一致法という高度な構文解析にもとづいた連文節変換を実現しています。

感性を刺激する驚異の表現力 高解像度自然色グラフィックス

●512ドット65,536色同時発色/クロームやチタニウムに代表される高品位な金属の質感、金・銀表現、人の眼に映る色や形状をほとんどありのままに表現し得る自然色グラフィックスが、これまでのC.G.イメージを一新します。

●1024×1024の実画面エリアを装備した高解像度表示能力/テキスト、グラフィックともに1024×1024ドットの実画面エリアをもち、最大表示エリアは768×512ドット(65,536色中16色指定可能)。専門分野にも対応できる表示能力です。未表示エリアへのスクロールも自在。

●未体験の動画が駆使できるスプライト機能/新開発のスプライトICを搭載、16×16ドットの緻密なキャラクタが1ドットごとのスムースな動きで、512×512ドットの高解像度画面を縦横に疾走する。クリエイターの感性を刺激する新しい能力です。最大表示は水平32スプライト、1画面128スプライト。色表示も65,536色中16色指定が可能です。

●凝似高解像度スーパーインポーズ/512×512ドット(インターレース方式)レベルのスーパーインポーズ。さらにオーバースキャン機能の採用でスーパーインポーズによるテロップ文字の不自然な切れもありません。

●テキストビットマップによるフレキシブルな画面設計/独立したテキスト画面を装備するとともに、グラフィック同様のビットマップ方式を採用(65,536色中16色指定可能)。テキスト画面をグラフィック画面としても活用できます。しかも両画面の重ね合わせ表示もできるフレキシブルな画面構成です。

■リアルなサウンドシーンをクリエイトできる8重和音ステレオFM音源搭載■肉声や臨場音、音楽までもメモリやディスクに音声ファイルとしてもつことができるサウンドデジタイズ記録AD PCM■オートロードやオートイジェクト、インテリジェントな機能を装備した1Mバイト5"FDD 2基搭載■操作のほとんどは手のひらで、狭い場所でも使える新開発マウス・トラックボール■ハードディスクはもちろん、イメージ入力端子、立体視端子など独自のインターフェイスを装備■3モードオートスキャン方式、高精細度カラーディスプレイテレビ(別売)。

※画面はハメコミ合成です。また、表示内容は実際とは多少異なる場合もあります。

あふれるクリエイティブマインド、いよいよ走り出したアプリケーション。

■ミュージック PRO 68K CZ-213MS 8月発売予定

メロディ譜、ピアノ譜、スコア(総譜)など、自由なレイアウトで書き込んだ譜面を内蔵のFM音源で演奏できる楽譜ワープロ&演奏用ミュージックツール。演奏データをBASICのPLAY文として出力も可能、サウンドPRO 68Kで作成した音色データも使えます。

■サウンド PRO 68K CZ-214MS 8月発売予定

1ファイル200音色単位で音色データの作成、編集が可能なサウンドエディティングツール。FM音源の全パラメータのグラフィックコントロールはもちろん、3D波形表示、言葉のイメージによる音色づくりができるイメージモードなど豊富な機能を装備しています。

■ビジネス PRO 68K CZ-212BS 8月発売予定

スプレッドシート機能、データベース機能、グラフ機能を合わせ持つ統合ビジネスソフト。9999行×255列の巨大なカルクシート、最大16個のウインドウ、充実の日本語入力、マウス対応のやさしいオペレーション。日本のビジネスシーンにふさわしいツールです。

●グラフィックツール「Z'S STAFF PRO 68K」(有)ツァイトより近日発売 ●統合型スプレッドシート「KAMIKAZE」(株)サムシンググッドより8月発売予定●ゲームソフト「ゼビウス」(電波新聞社)、「スペースハリアー」(電波新聞社)、「レリクス」(ボーステック(株))…以上近日発売

X1turbo シリーズ用グラフィックツール turbo Z'S STAFF ジーズスタッフ

X1ターボシリーズの優れたグラフィック機能を存分に発揮させる待望の本格グラフィックツールです。カラーイメージボード、スーパーインポーズなどの独自機能にも対応。ペン・ブラシ・ペイント・パレット・拡大縮小など多彩な作画機能、各種文字フォント(標準・斜体・縁どり・影つき・下線・サイズ)を装備。キーボードはもちろんマウスやジョイスティックによる簡易入力も可能です。400ラインモード対応。

■2D・5"FD版 CZ-137SF 標準価格19,800円

NEW X1/X1turbo シリーズ用グラフィックツール X1Z'S STAFF ジーズスタッフ

ターボ・ジーズスタッフの高機能がX1でも…ユーザー待望のC.G.ツール。もう、ブラウン管をキャンバスがわりに思う存分アートする、クリエイティブなグラフィックの世界がどんどんひろがります。日本語入力にも対応。

■2D・5"FD版 CZ-138SF 標準価格 13,800円

NEW X1turbo シリーズ用 グラフィックライブラリー

Z'S STAFFや嬉楽画ターボ、嬉楽画で使用可能なデータ集です。3枚のディスクの中には、年賀状、クリスマスカードをはじめ利用価値の高いイラストやPOP文字がデータとしてつまっています。入力はキーボード、マウス、ジョイスティックをサポート。X1ターボシリーズのグラフィック世界がさらにひろがります。

■2D・5"FD版 CZ-140SF 標準価格9,800円

SHARP

X1 turbo シリーズ用 コスモステーション

X1ターボシリーズをホストマシンとしてホスト局を運営するためのソフトウェアです。パソコンシーンに新しい分野をひらく「パソコン通信」、既に全国各地で大小さまざまなネットワークが展開され、参加者も増加の一途をたどっています。コスモステーションは、そうしたアクセスするだけの通信ではなく、あなたのターボをホスト局に、あなたの住む街でBBSや電子メールなど、パソコン仲間が気軽に話せるミニ通信基地を築くためのソフトです。

▶ホスト局開設に必要なシステム
●X1turbo モデル30、X1turboII、X1turboIII、X1turbo Zのいずれか●モデムまたはモデムホン(CZ-8TM1他6機種対応) ●公衆電話回線(1回線) ●コスモステーション ●プリンタ(必要に応じて)

■「コスモステーション」による ホスト局仕様概要

仕様＼システム	2D・FDシステム	2HD・FDシステム	HDシステム
登録会員数	70人	128人	299人
メールボックス数	70	128	299
メール量	4,000文字	4,000文字	12,000文字
BBS1保存期間	10日	30日	30日
BBS2タイトル数	10タイトル	60タイトル	125タイトル
インフォメーション数	15ファイル	60ファイル	225ファイル
プログラム数	5ファイル	60ファイル	125ファイル

●X1turboモデル30、X1turboIIでの2HD・FDシステムにはフロッピーディスクユニットCZ-520Fが必要です。
●HDシステムにはハードディスクユニットCZ-500Hが必要です。

■2D・5″FD版 CZ-136SF 標準価格9,800円

X1/X1 turbo シリーズ用 モデムターミナル

モデムボードを同梱していますので、家庭でご使用中の電話に接続するだけで手軽にパソコン通信が楽しめます。各種ネットワークにも簡単にアクセス。またX1 turboシリーズユーザーによるBBSネットワークも構築できます。

■2D・5″FD版 CZ-133SF 標準価格25,800円 (モデムボード付)

X1 turbo シリーズ用 turboターミナル

各種ネットワークにアクセスしたり、パソコン通信(漢字対応)がスピーディに楽しめる通信ソフトです。

※公衆回線を使って通信する場合、モデム付電話か音響カプラが必要です。

●別売RS-232CケーブルCZ-8LM1(平行接続型) CZ-8LM2(クロス接続型) 各標準価格7,200円

■2D・5″FD版 CZ-131SF 標準価格8,800円

X1/X1 turbo シリーズ用 ミュージッククリエイタ NEW ミュートピア

ミュージッククリエイタ「ミュートピア」は、楽符を見ながら音符を入力していくという従来のミュージックツールとは異なり、マウス、ジョイスティックやキーボードを使ってパソコンを楽器に変えて演奏が楽しめるユニークなソフトです。五線紙ではなく、音の高低・長短を書き込んだグラフをもとに自動演奏。音符が苦手な人でも、画面を見ながらの簡単操作で作曲演奏が楽しめます。FM音源を強力にサポートした新しいミュージックシーンが体験できます。
●ワールド・マップモードでは、画面に世界地図が表示され、世界各地の民族音楽や代表的音楽ジャンルのデータ21個の中からセレクトして演奏できます。
●リズムもグラフ入力で行い、編曲の理論を知らなくても独自の編曲が可能です。

※ご使用に際してはターボZを除いてFM音源ボード(CZ-8BS1)が必要です。

■2D・5″FD版 CZ-139SF 標準価格12,800円

X1 シリーズ用 X1 LOGO

人工知能言語として注目を集めているLOGOがX1シリーズで走ります。基本的なLOGOの機能に加え、サウンド、マルチタートル機能をサポート。使いやすいBASICライクなスクリーンエディット機能やリスト処理機能も備えています。

■2D・5″FD版 CZ-134SF 標準価格9,800円

X1 turbo シリーズ用 turbo LOGO(漢字版)

プロシジャー名や変数名の他、ワードやリストの中でも漢字が使えます。また本格活用に応えるスピードとノード数(約5,000)を確保。マルチタートル、シェイプ、マウス、音楽機能もついた多機能ぶりです。あなたの知的創造の世界がさらに拡がります。

■2D・5″FD版 CZ-117SF 標準価格18,800円

X1 turbo シリーズ用 turbo CP/M® V2.2(漢字版)

X1ターボ特有のハードをサポートするとともに、ビジネスユースに欠かせない日本語処理機能も付加。WORD MASTER™も搭載。

■2D・5″FD版 CZ-130SF 標準価格14,800円

X1/X1 turbo シリーズ用 ランゲージマスター(CP/M®)

オペレーティングシステムCP/Mがさらに手軽に。便利なスクリーンエディタWORD MASTERもついています。

■2D・5″FD版 CZ-128SF 標準価格 9,800円

X1/X1 turbo シリーズ用 ランゲージシリーズ

■各2D・5″FD版 各標準価格13,800円

科学技術計算の分野に適した高級言語。使いやすいトレーススタイルのデバッグが可能です。
FORTRAN (CZ-115LF)

いま熱い視線を集めるC言語。Cコンパイラとして定評のBDS C Compilerのサブセット。
C (CZ-116LF)

事務分野で威力を発揮する伝統の言語。有効桁数やファイルの定義、データ転送が容易。
COBOL (CZ-118LF)

人工知能研究の中心的言語。効率の良いリスト処理が特長です。
LISP (CZ-120LF)

拡張性に優れたスクリーンエディット型言語。とくに適用分野を選ばない自己増殖型言語です。
FORTH (CZ-120LF)

系統的プログラミング設計に適した言語。初めてプログラムを学ぶ人にも最適です。
PASCAL (CZ-125LF)

文法が明快な数学的プログラミング言語。すべての操作を関数の集まりで表現できます。
APL (CZ-126LF)

ランゲージシリーズの使用にあたっては、CZ-130SF、CZ-128SF、またはCZ-5CPMが必要です。CP/Mは米国デジタルリサーチ社の登録商標です。WORD MASTERは米国マイクロプロ社の登録商標です。

X1シリーズ用 NEW BASIC (Version 2.0)

■カセット版 CZ-112SF 標準価格7,800円
■2D・3″FD版 CZ-113SF 標準価格8,800円
■2D・5″FD版 CZ-124SF 標準価格8,800円

シャープ株式会社●お問い合わせは…シャープ㈱電子機器事業本部テレビ事業部 第4商品企画部 〒162東京都新宿区市谷八幡町8番地 ☎(03)260-1161(大代表)へ。

資料請求券 CZソフト 8体

X1・X1 turboシリーズ用 周辺機器

C.G.や映像の高画質フルカラープリントを実現するビデオプリンタ。

パソコンやビデオ機器に対応。64階調(485×480ドット)で再現する昇華性染料熱転写方式を採用。

中間色も自然ななめらかさで再現、深みのある色調のカラープリント。イメージ豊かなC.G.やカラフルな映像が鮮やかに残せます。

●標準、拡大のどちらでも選択できる2種類のプリントサイズ(ビデオ信号入力時)●左右反転プリント可能●用途に応じて高画質な白黒プリントが可能●ビデオ入力端子、アナログRGB入力端子、デジタルRGB入力端子、パラレルインターフェイス、と豊富な入力端子で各種映像情報機器に対応●画像や文章を手軽にプリントできるX1/X1turboシリーズ用ソフトウェア〝カラープリントツール〟を同梱。

カラービデオプリンタ NEW

CZ-6PV1……………………標準価格 198,000円

●パーソナルコンピュータ及びディスプレイは別売。画面はハメコミ合成です。

イメージ豊かな映像表現、立体映像も楽しめる。感性あふれるアートツール。

テレビ・ビデオ映像をカラー静止画に――。

カラーイメージボードII NEW

CZ-8BV2……………標準価格 39,800円

●画像処理ツール、およびグラフィックソフト「嬉楽画」・「楽々ぽっぷ漢単」を同梱。取り込んだ画像を自在に修正・加工できます。

パソコンで初めて立体映像を実現――。

立体映像セット

CZ-8BR1……………標準価格 29,800円

X1/X1ターボシリーズと組み合わせて迫力あるフルカラー立体映像が手軽に楽しめます。立体作画ソフトも装備。立体エアチェックやイメージ処理も。

C.G.のハードコピーも美文書も。第2水準漢字ROM搭載。

熱転写カラー漢字プリンタ NEW

CZ-8PC2
標準価格 69,800円

C.G.はもちろんカラーイメージボードで取り込んだ映像も鮮やかにカラープリント。文書作成にも24×24ドットの高品位印字で対応。

システムづくりに応える多彩な周辺機器群(価格は標準価格)

プリンタ		
●24ピン漢字プリンタ(80桁)	CZ-8PK5	129,000円
●24ピン漢字プリンタ(136桁)	CZ-8PK6	159,000円
●ドットプリンタ	CZ-8PD3	59,800円

FM音源		
●ステレオタイプFM音源ボード	CZ-8BS1	23,800円

※スピーカ(2本1組)標準装備、ミュージックツール同梱

ファイル装置		
●ミニフロッピーディスクユニット(2HD・2DD) ※1	CZ-520F	118,000円
●ミニフロッピーディスクユニット(2D)	CZ-502F	99,800円
●ミニフロッピーディスクユニット(2D・1ドライブ)	CZ-503F	49,800円
●ハードディスクユニット(10MB)	CZ-500H	348,000円
●増設用ハードディスクユニット(10MB)	CZ-501H	258,000円

●カセットデータレコーダ	CZ-8RL1	24,800円
●ミニフロッピーディスク	CZ-5M2D/CZ-5M2HD	(各10枚入)
●コンパクトフロッピーディスク	CZ-3FBD	1,300円

ビデオ編集装置		
●パーソナルテロッパ ※2	CZ-8DT2	44,800円

拡張ボード・その他		
●320KB外部メモリ	CZ-8BE2	29,800円
●RS-232C・マウスボード ※3	CZ-8BM2	19,800円
●JIS第1水準漢字ROM ※4	CZ-8BK2	19,800円
●JIS第2水準漢字ROM ※5	CZ-8BK4	6,800円
●JIS第2水準漢字ROM&ターボ博士レキシコン・日本語百科ワードパワー ※6	CZ-8BK3	13,800円
●フロッピーディスクインターフェイス ※7	CZ-8BF1	14,800円

●RS-232C用ケーブル(平行接続型)	CZ-8LM1	7,200円
●RS-232C用ケーブル(クロス接続型)	CZ-8LM2	7,200円
●拡張I/Oポート ※8	CZ-8EP	11,800円
●拡張I/Oボックス	CZ-8EB3	33,800円
●RFビデオコンバータ ※9 ★	CZ-8VC	15,800円
●RFコンバータ ※10	AN-58C	2,980円
●モデムユニット(300ボー)	CZ-8TM1	29,800円
●モデムユニット(300/1200ボー自動切換)	CZ-8TM2	49,800円
●マウス	CZ-8NM1	13,800円
●チルトスタンド ※10	CZ-6ST1(B・E)	5,800円
●チルトスタンド ※11	CZ-81T(S・R)	8,500円
●システムスタンド	CZ-8SS2	5,500円
●ジョイカード	CZ-8NJ1	1,700円

●品番中の( )表示は、S〈メタリックシルバー〉・R〈ローズレッド〉・E〈オフィスグレー〉・B〈ブラック〉を示します。※1 X1ターボシリーズ用 ※2 CZ-862Cには接続できません。※3 X1シリーズ用 ※4 CZ-802C, 803C, 811C, 820C用 ※5 CZ-856C用 ※6 CZ-850C, 851C, 852C, 862C用 ※7 CZ-850CでCZ-520Fを使用する場合、またCZ-803C, 804C, 811C, 820C, 850CでCZ-300Fを使用する場合に必要 ※8 CZ-800C, 802C用 ※9 CZ-862Cには接続できません。※10 CZ-600D, CU-15M1用 ※11 CZ-801D, 802D, 811D, 850D, 855D, 870D用 ★在庫僅少 ●接続等の詳細につきましては、周辺機器総合カタログをご参照ください。

シャープ株式会社 ●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
電子機器事業本部テレビ事業部第4商品企画部〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

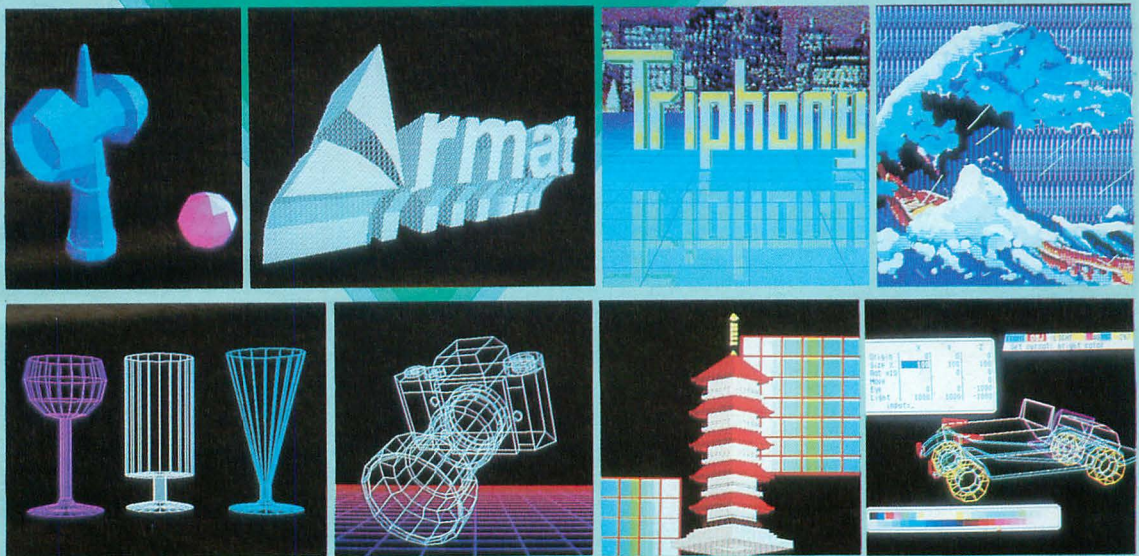
資料請求券 X1・X1turbo 周辺機器

本格派立体映像

三次元グラフィックスソフトウェア

Triphony トリフォニー

¥16,800
X1turboシリーズ用 5インチ2D版

トリフォニーは全く新しい三次元コンピュータグラフィックスシステムです。簡単な操作で立体イメージをディスプレイ上に作成することができます。立体映像セットを利用すると奥行きを確認しながら立体を作成していくこともできます。
デザイン、シミュレーション、CG・CAD教育用として最適です。

トリフォニーシステムは三次元処理を行なう「3Dモデラー」と、手描き用「ペイント」の2種類のソフトウェアから成り立っています。

3Dモデラー
3Dモデラーはコンピュータグラフィックスの基本的な表示モデルである、ワイアーフレーム・サーフィスモデル(単色)・レンダリングモデル(カラー)の3種類をサポートします。立体はrotate(回転体作成)、sweep(面厚み付け)などの立体構成コマンドにより簡単に作成できます。作成した立体には、shadeコマンドによって美しい陰影(シェード)を付けてレンダリングすることが可能です。

立体映像セット(CZ-8BR1)
ペイント
ペイントは3Dで作成された画像に修正を加えたり、着色したりすることができます。勿論、すべて手描きで画像を作成することも可能です。バックグラウンドモードの採用により、透明感・光沢なども表現できる高度な描画機能を持っています。

トリフォニーの機能概要

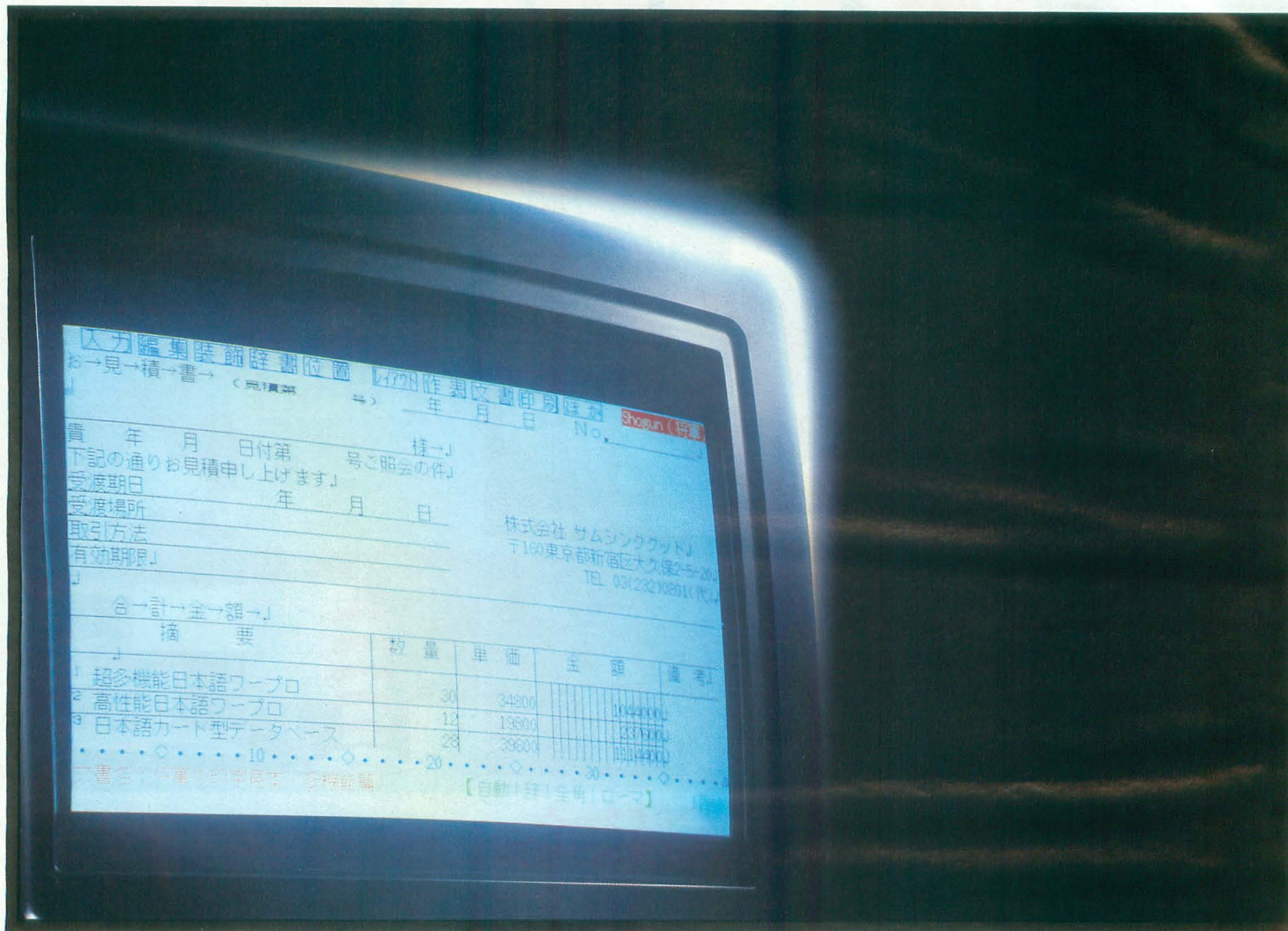
解像度：	モノラル 640×400（高解像度） モノラル 640×200（高・低解像度） 立体モード640×200（高・低解像度）
必要機器：	マウス・2ドライブ（1MBタイプにも対応）
対応機器：	立体映像セット(CZ-8BR1)・320KB外部メモリ カラーイメージボード カラープリンタ(CZ-8PC1)・各種白黒プリンタ
マニュアル：	約200ページ
3Dの機能：	正面図・上面図・側面図表示、拡大縮小・回転・移動など座標変換機能、パースオンオフ、グリッドオンオフ、シェード(陰影付け)、スクウェア・サークル・ローテート(回転体)・スウィープ(厚み付け)・ハイド(隠面処理)・ハードコピー・ヘルプその他ファイルアクセスコマンド等レンダリング機能(最大2500ポイントまたは500ポリゴン)
ペイントの機能：	セット・フォアグラウンド・バックグラウンドモードによるブラシ・ライン・ボックス・ボックスフル・グラデーションボックス・コピーなどのファンクション、フィル・エッジ・拡大縮小・画像入力(turboZ以外はカラーイメージボード要)

■「トリフォニー」は全国の有名パソコンショップなどでお求め下さい。通信販売をご希望の場合は現金書留または郵便振替で当社までお申し込みください。(送料当社負担)

(有)アーマット
〒227 横浜市緑区荏田町473-5
TEL:045-911-7427

<トリフォニー開発に利用されたソフトウェア>
(1) Z80アセンブラ開発セット MR−ASM・MR−1D 12,800円
(2) BDS Cユーティリティパッケージ 10,000円
(1)は有名パソコンショップで、(2)は通信販売でお求めください
(郵便振替 横浜5-30518) (有)アーマット

資料請求券 Oh! MZ

本物の条件。

16ビット用最新、自動/一括/連文節変換システムKatana(刀)の完全移植。143万通にも及ぶ多彩な文字表現。[※1]本格的データベース、表計算機能搭載。16ビットワープロソフト、データベースソフトなどMS-DOS上で動くソフトとのデータ互換。[※2]その他すべての機能が16ビット用に開発されたパーツ群により構成。フルスペックでなおかつ超高速。

※1.文字サイズ・文字種・文字の位置・網かけ・下線・カラー設定の組みあわせによる計算。
※2.MS-DOSとのデータ交換は2HD版のみ。※MS-DOSはマイクロソフト社の登録商標です。

●一発で変換できます。

●印刷イメージを表示するリアルモード

●全体のイメージを見る縮小モード

●印刷書式は細やかな設定が可能

●再計算までOKの表計算機能

●用途多彩なデータベース機能

Katana(刀)が自動・一括・連文節変換実現。

サムシンググッドが16ビット機上で開発した変換システムKatana(刀)を8ビット機用にコンバート。8ビットで初めて自動変換・一括変換・連文節変換を可能にしました。上段左端の写真のような文章も一気に漢字かなまじり文に変換します。しかもKatana(刀)の大きな特長は、品詞分類のきめ細かさと、独自の評価点数法を確立したこと。品詞をこれまでの倍以上(当社比)に分類し、かつ文節と文節のつながり方の妥当性を評価点によって判定することにより、既存の16ビットワープロソフトにも勝る高い変換効率を誇ります。

SGソフトウェアライブラリー

縦1倍・横2倍(横倍角)文字

縦½・横1倍文字

¼角上つき

均等割付機能

反転文字＋部分細密機能

下線は全部で31種。打消し、文字上に入れられます。

縦方向の倍角文字、½倍文字は、位置を3通りに変えられます。

全角文字

縦1倍・横½倍(半角)文字

罫線は8種。文字間、文字上ともに引くことが可能です。

表計算の結果をそのまま文書に使うことができます。

デジマルタブ機能

網かけは全部で31種。枠囲いもあります。

カラーは白色(印字しない)を含む8色を設定可能です。
※カラー印刷はプリンターによってはできないものもあります。ご注意ください。

いよいよ発売間近

カード型データベース機能、表計算機能搭載。

住所録、名刺管理、カセットライブラリーなど使いみちタップリのデータベースと、行間・列内・列間と多彩な計算が可能な表計算機能を搭載。

他の追従を許さぬ文字表現力。

文字サイズは、1/4角から横4倍縦2倍角まで15種類。すべてのサイズの文字を、強調文字、白黒反転文字、斜体文字、袋文字に変換することが可能。これらの機能は、漢字・かな・記号など文字の種類を問いません。

多様な用紙への印刷が可能です。

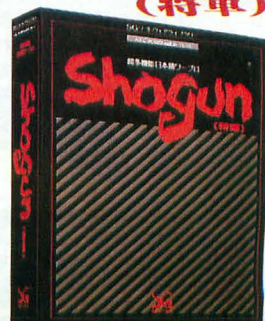
はがき、原稿用紙、タックシールへの印刷を簡単に行うために専用の用紙設定を用意いたしました。

SHARP X1 turbo III / Z 専用2HD版
SHARP X1 turbo シリーズ対応2D版
※本商品はX1ではお使いいただけません。あらかじめご了承ください。

2D版、2HD版ともに ¥34,800 ※本商品はX1ではお使いいただけません。あらかじめご了承ください。

※ Shogun(将軍)の画面デザイン・仕様等は改良を目的に予告なく変更する場合がございます。あらかじめご了承ください。
※ Shogun(将軍)は、フロッピーの種類およびハードウェアのメモリ容量によって機能に違いがあります。あらかじめご了承ください。
〈即戦力Samurai〉〈即戦力〉をお使いの方はShogun(将軍)へのシステムアップサービスがございます。くわしくは弊社営業部までお問いあわせください。

〒160東京都新宿区大久保2-5-20シティプラザ新宿　TEL03(232)0801(代表)

※資料のご請求は右の券を切りとり左記の弊社営業部宛までお送りください。カタログ等お送りいたします。

資料請求券 Oh!MZ Ⓢ 8月号

©1986 NAMCO, ALL RIGHTS RESERVED

リターン オブ イシター

X-1ターボ専用開発順調!

あのテーブルゲームの「リターンオブイシター」がパソコンにのって登場です。
「ドルアーガの塔」でドルアーガを倒した塔の最上階よりスタートし、128ある部屋をめぐって塔を脱出すればゲームは終了です。(全部の部屋を回らなくても、脱出は可能です。)
呪文も64種で早く覚えて有効に使いましょう。でも、最初は全部使えません。全部使いこなせれば、128面クリア保証です。ゲームは簡単/謎は難解/

●この写真は開発中の画面です。

●対応機種●
PC-8801シリーズ(SR以降)、PC-9801シリーズ、X-1/turboシリーズ

全機種FM音源対応! 価格6,800円

真の1200ボーのスピード、ご存知ですか?

X1 turbo 専用パソコン通信ソフト(モデル10を除く)

日本語入力は文節変換。
フロントプロセッサにJET-CORE™を採用。JET-X1の文書もO・K!!

BASIC JET-X1
JET ターボ ターミナル
ファイル転送 MS-DOS

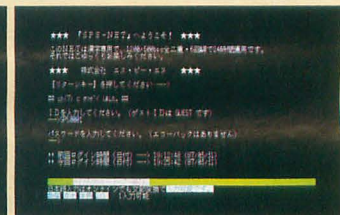
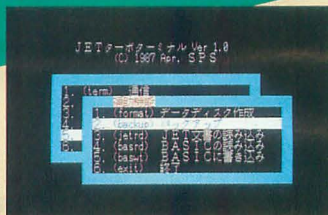
JETターボターミナル

150〜9600bps対応

オートダイヤル、オートログイン、アップロード、パラメータの設定
エスケープシーケンス対応、ファイル管理機能、フルスクリーンエディタ搭載。

対応モデム:VM12、CZ-8TM1、CZ-8TM2、SR-120AT等

好評発売中5インチ2D1枚 SS-101**9,800円**

ファイル管理メニュー

ターミナル画面

戦慄の アドベンチャー

Re Birth
リ・バース

好評発売中!

失われた記憶を取り戻すために
謎の城へ足を踏み入れた。
そこで主人公が見たものは…。

リ・バースは1Mバイトをこえる超大作だ!!

GS 101	X-1/turbo シリーズ	5"2D 4枚組	¥7,800	全機種 カラーモニター フロッピーディスクドライブ (2ドライブ) 漢字ROMが必要です。
GS 102	PC-8801 シリーズ	5"2D 4枚組		
GS 103	MZ-2500 シリーズ	3.5"2DD 2枚組		
GS 104	FMシリーズ	3.5"2D 4枚組		
GS 105	FMシリーズ	5"2D 4枚組		
GS 106	PC-9801 シリーズ	3.5"2DD 2枚組		
GS 107	PC-9801 シリーズ	5"2DD 2枚組		
GS 108	PC-9801 シリーズ	5"2HD 2枚組		

FM音源対応
(PCはSR以降 X-1はCZ-8BS1, MZ2500 FMシリーズ, PC9801シリーズ)

㈱マイコンハウス SPS
〒960 福島市太平寺字堰ノ内5-3 ☎(0245)46-5777
FAX(0245)45-1804(GII,GIII)

まじめに将棋の勉強を。

棋太平は、単にコンピュータと対局するだけでなく、名人戦の設定再現・駒落ち対局それらの記録再現なども自由自在です。

現在、下記のパソコンが使用できます。

GS 051	X-1/turbo シリーズ	5FD ¥6,500	CZ-800は、要G-RAM カラーモニター使用 フロッピーディスクドライブ並びにデータレコーダは、純正品のみ動作確認済み ジョイスティック対応 純正マウス対応
GS 052	X1/turbo シリーズ	CT ¥4,500	
GS 053	MZ-2200/2000 シリーズ	5FD ¥6,500	MZ-2000は、要G-RAM1、2、3、グリーンモニタ使用可 フロッピーディスクドライブ並びにデーターレコーダーは、純正品のみ動作確認済み
GS 054	MZ-2200/2000 シリーズ	CT ¥4,500	
GS 055	PC-8801 全シリーズ	5FD ¥6,500	カラーモニタ使用 フロッピーディスクドライブ並びにデータレコーダは、純正品のみ動作確認済み アスキーマウス対応
GS 055	PC-8801 全シリーズ	CT ¥4,500	
GS 057	MZ-2500	3.5FD ¥7,000	カラーモニタ使用 ジョイスティック対応 純正マウス対応
GS 061	FM7/77/AV	3.5FD ¥7,000	カラーモニタ使用。フロッピーディスクドライブ並びにデーターレコーダーは、純正品のみ動作確認済み ジョイスティック対応 純正マウス対応
GS 062	FM7/77/AV	5FD ¥6,500	
GS 063	FM7/77/AV	CT ¥4,500	

当社の製品は全国の有名デパート、パソコンショップでお求めになれます。尚、お求めになれない場合、郵便局にてお申し込みください。●口座番号 郡山5-12298 ●加入者名㈱エス・ピー・エス ●金額 代金合計 ●通信欄(裏面)ご希望ゲームソフト名、数量、代金合計、年齢、氏名、機種名、テープかディスクの種類。(一週間以上かかりますので、お急ぎの方は現金書留をご利用ください。その場合、おつりのいらないようにお願いします。)

パートナーショップ
キャリーラボ マイクロキャビン

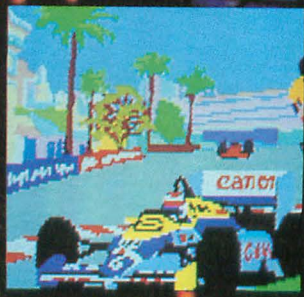
ユーザー待望のワールドマップソフト

WORLD ings 169

遊んで学ぶ

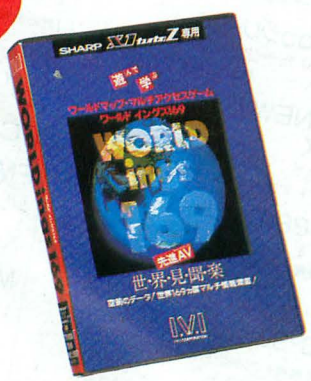
追跡・推理ゲ

地理
文化

全世界169ヵ国A・V(オーディオ・ビジュアル)体験。きわめれば君は世界の情報通。

●世界を巡るマルチアクセスゲーム
日本と関係深い某国間の機密を収められた小型ICカードが何者かによって、日本国外に持ち出された。このICカードを奪回すべく、日本をスタートに各国情報局からの調査データをベースに推理をしていく追跡ゲーム。犯人の手掛りを得るため、世界の国々169ヵ国を訪問、歴史的な名所旧跡をたずねて調査、分析をするうちに世界の地理、国々の文化情報の「通」になり、事件解明への推理感覚のボルテージがアップされ、見事ICカードを奪回した時のキミは、世界地理文化のオーソリティ——。

●世界地理文化の検策機能(国めぐり) マニュアルによるキー操作で、世界169ヵ国のデータを検策。グラフィックス、文字情報を呼びだし、世界地理としても活用できます。

★主要国の国歌15曲と地域の民族音楽6曲のFMステレオ演奏(turboZのみ)
★世界地図のフルカラーグラフィックス6画面
★世界169ヵ国別フルカラーグラフィックス169画面
★世界169ヵ国別、国勢文化の文字情報35,000余語

好評発売中

ワールドマップ・マルチアクセスゲーム
WORLD ings 169
ワールド イングス169

SHARP X1 turbo Z 専用(5″2HD版2枚組) ¥12,500
X1 turbo II 専用(5″2D版5枚組) ¥11,000

●X1 turbo II 専用は、turbo II・turbo III・turbo(2ドライブが必要)に対応します。

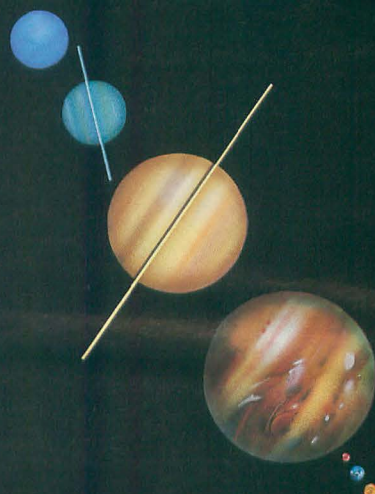
発売元
I.V.I
アイ・ヴィ・アイ株式会社
大阪市浪速区日本橋5丁目21-22 郵便番号556
電話番号 (06)631-2867(代)

告映像は、3次元CGで制作したものです。3次元CGのご用命は当社まで。

通信販売のごあんない
お近くのショップで手に入らない方、手に入りにくい方は、通信販売をご利用ください。ご希望の方は、①使用機種名 ②商品名 ③住所 ④お名前 ⑤電話番号を明記のうえ、代金を現金書留で弊社にお送りください。(送料

ARTDINK

SPACE ROMAN

本格的SF シミュレーション ゲーム

地球防衛軍

私の星を守りたい

本格的SFシミュレーションゲーム「地球防衛軍」は
速度を出すためのメインエンジン
左右への旋回能力を決めるサブエンジン
敵を攻撃するための陽子砲
戦闘機を発見するための測距レーダーなど
地球を守るために味方戦闘機に搭載できるものは多彩です。
「地球防衛軍」の総司令官であるあなたは
各プロセッサに戦略・戦術をインプット。
あなたの設計した戦闘機と敵戦闘機が
7つの惑星を舞台に
ストラテジックな戦闘を展開します。
このゲームの勝敗を左右するのは
総司令官の手腕次第。
果たして勝利の女神は
微笑んでくれるでしょうか。

本格的SFシミュレーションゲーム

地球防衛軍

X-1turboシリーズ
●7月25日発売予定
【メディア】5"2D 【価　格】7,800円

FM7・NEW7・77・77AV シリーズ
●近日発売
【メディア】5"2D・3.5"2D 【価　格】7,800円

PC-8801シリーズ
●新発売
【対応機種】PC-8801mkII SR/FR/TR/MR FH/MH/PC-88VA
【メディア】5"2D 【価　格】7,800円

PC-9801シリーズ
【対応機種】PC-9801E/F/M/VF/VM/UV/VX VM21/XL
※PC-9801及びPC-9801Uでは作動しません。
※必要メモリ256KB
【メディア】5"2DD 5"2HD 3.5"2HD
【価　格】9,500円
※FM音源対応

本格的鉄道シミュレーションゲーム

A列車で行こう

PC-9801シリーズ
【対応機種】PC-9801/E/F/M/VF/VM/U/UV VM21/VX/XL
※必要メモリ256KB　※FM音源対応
【メディア】5"2DD 5"2HD 3.5"2DD 3.5"2HD
【価　格】9,500円

PC-8801シリーズ (5"2D)

FM7・NEW7・77・77AV シリーズ (5"2D・3.5"2D・テープ)

X-1turboシリーズ (5"2D)

MZ-2500シリーズ (3.5"2DD)

【価　格】ディスク7,800円　テープ6,800円 (FMのみ)

ARTDINK

株式会社 アートディンク
〒275 習志野市津田沼2-11-20
TEL.0474-77-7541 FAX.0474-78-6280

●お求めは、お近くのパソコンショップ、または現金書留にて（送料サービス）

THE SOFTOUCH SPECIAL

Z'sSTAFF PRO 68Kの世界

ついにその全貌を現したZ'sSTAFF PRO 68Kを2回にわたって完全紹介する。まず今回は基礎的な描画機能を中心とし，次回には編集および特殊効果機能へと展開する。X68000でなければできないイメージプロセッサとしての可能性を追究してみたい。

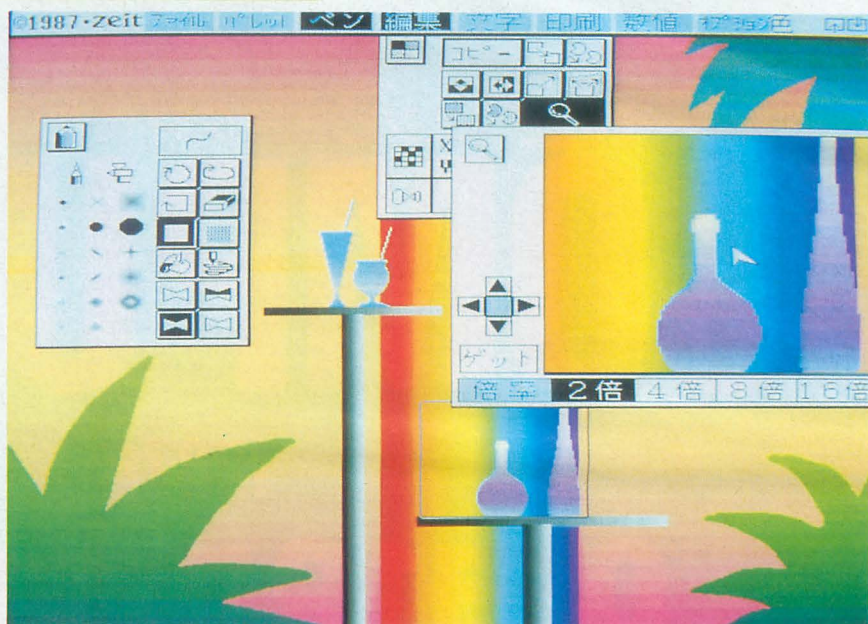
Saito Susumu
斎藤 晋

このZ'sSTAFF PRO 68Kを紹介するのは大変なことなのだ。なにしろあまりにも度を超えた表現力に，あのログイン誌（7月号）でも「もうZ'sSTAFFなんて呼べない」と銘打って華々しく紹介されている(まだまだ半分ぐらいしかできていない段階での話だ)。実際にはZ'sSTAFFに「PRO」が付くことになったわけだが，それで解決のつく問題ではない。

私はこのようなマシンでこのようなソフトが利用できる時代をずうっと待っていた。正直にいおう。私はFM77AVというマシンを持っている。私はSMC-777Cというマシンも持っている。4096色は私の青春であった。しかし，さらに告白すれば，私は編集室の98でZ'sSTAFFを使うためにマウスを買ってしまったことだってある。私が25,000円払って買ったそのマウスはいまや「上海」専用となり，98ともども編集室の福利厚生施設として役立っているが，そんなことはどうでもいい。私がZ'sSTAFFを使っていたのは，たった8色でこんなにも多彩な表現が可能になるのか，といった驚きがあったからだ。もしこれで色数が増えたなら，いったいどんなグラフィックツールが登場するのだろう。私の期待はいやがうえにも盛り上がってしまったものである[注1]。

Z'sSTAFF PRO 68Kはまだ開発途中の段階にある。本来なら完全な商品バージョンになってからきちんと評価すべきところだが，X68000ユーザーにそんな余裕はない。それに現段階でZ'sSTAFF PRO 68Kはすでに評価を超える恐るべきパワーを発揮しているのだ（どうだ，ワクワクしてきただろう）。

システムの概要

すごいすごいとばかりいっていてもしようがないので，ともかくZ'sSTAFF PRO 68Kのシステム概要からお伝えしよう。できることのスケールは段違いだが，基本的な操作の手順などは従来のZ'sSTAFFと変わらない。システムを起動させるとまず画面上部にいくつかの項目が表示されたメニューバーが現れる。ここから目的の項目を選んでマウスの左ボタンをクリックすると，その項目に関するさまざまな機能が並んだウィンドウが開かれ，さらに機能によってはその下にまたウィンドウが用意されている，といったぐあいに階層構造をなしているわけである。

基本のメニューバーには，ファイル，パレット，ペン，編集，文字，印刷，数値，オプションの8つの項目，それに現在選択されている色と画面スクロールのアイコンが表示されている。このテのソフトにちょっとでも触ったことのある人ならば話は早い。たいていの人はさっさとパレットで色を選択し，ペンで適当な作画機能を選んでおもむろに描きなぐるのである。編集は絵の加工・修整，文字は当然日本語得意？ で，仕上げの印刷といった大まかな流れがおわかりだろう。

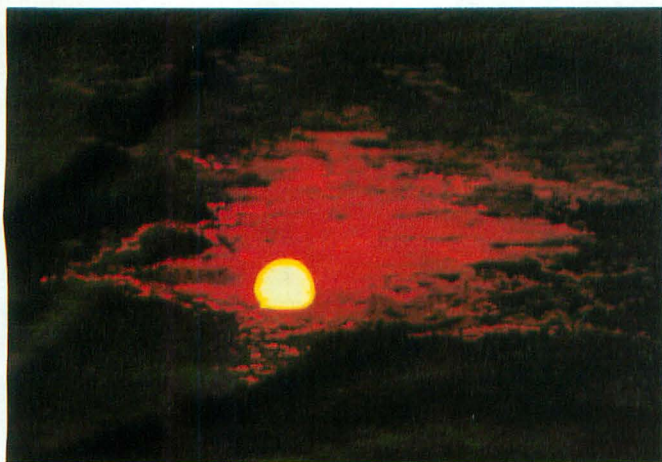
以下に基本メニューの概要をざっと紹介しよう。パレットとペン以外はまだ仕様が固まっていないので正確なところはお伝えできない。あくまで概要と思ってあたたかく見守ってほしい。

ファイル

いうまでもなく絵のロード/セーブを行うところだ。が，絵だけではない。このあと紹介するもっとも美味しくて驚異的な部分，パレット，タイル，トーン，そしてペンも，それぞれ独立して保存できる。

画面は通常の1画面のほかに縦2画面分の大きさの絵をエディットすることもできる（メニューバーの右端にあるアイコンをクリックすると画面が少しずつスクロールする)。また，任意の領域を部分セーブすることによって画面の切り貼り合成も可能となる。

パレット

自由自在に色を作ったりストックしたりできるのがこのウィンドウだ。スクロールするカラーサンプル，驚異のグラデーション機能，画面の任意の場所から色を拾ってくるスポイト機能，本物のパレット同様に色を混ぜ合わせることだってできるのだ。さらにサブウィンドウとして，HSV(色相，彩度，明度）によって色を選ぶ円形のカラーチャート，色の代わりにさまざまなパタ

階調のついたペンのおかげで，いままでのグラフィックツールでは不可能に近かった雲のようなものも簡単に描ける。まだ慣れていないのでペンを取っかえひっかえ試してみたがちゃんと夕日に見えるだろうか。なお，こういった同系色の絵はグラデーションで色を選ぶとラクチン。

こちらは夕日と違って10分もあれば描けちゃう例。細かい部分はルーペを使用。背景のグラデーションも一瞬だ。

ーンを利用できるタイル，そしてこれまた圧倒的な威力を発揮するスクリーントーンもある。いずれの機能もX68000級だ。

ペン

あらゆるタッチを可能にするペン先のエディット機能と，豊富なラインおよびペイント機能がZ'sSTAFFの魅力。このX68000版ではペンの形だけでなく階調までもエディットすることができ，水彩画のような淡い表現やにじんだ表現なども可能となる。ペンやエアブラシの機能としては，直線，フリーハンドによる2種類の曲線，スプライン曲線，円，楕円，ボックス。ペイント機能としては，境界色内ペイント，領域内ペイント，ボックスフルなどがある。

さらにさらに，Z'sSTAFF PRO 68Kには便利なマスキング機能がついているのであった。よしよし。

編集

編集のウィンドウには，コピー，ムーブ，拡大/縮小，上下反転，左右反転，ルーペ，といったなんとなくお馴染みの機能はもちろん，X68000ならではのカラーチェンジ，それにエフェクト機能としてモザイク変換，ソフトフォーカスなどができるようになっている。

文字

ちょっとしたコピーから本格的な解説文まで，画面に重ねて表示することができる。日本語の入力には標準の日本語フロントプロセッサASK68kを利用している。影つき，袋文字，斜体などは当然，アルファベットには数種類のフォントがあらかじめ用意される予定だ。

印刷

各種プリンタをサポート。カラーあるいはモノクロのハードコピーが取れる。

数値

図面のような正確さを要求されるものを描く場合に有効な機能が用意される。たとえば，線を1本引くにしても線分の長さを指定したり，座標を表示させたりといったことができる。またここでは，マウスの動きの速さを変えたり，ダブルクリックの間隔を自由に設定することもできる（編集室に届いたサンプル版はダブルクリックの間隔が異常に短く設定されていた。ツァイトの人はものすごくダブルクリックが速いに違いない）。

オプション

Z'sSTAFF PRO 68Kでは，外部からのイメージ取り込みに関しても強力なサポートがなされている。まず，イメージスキャナはIN501/502，IN503（日本電気），GT-3000（エプソン）が利用できる。IN501/502はモノクロ単階調，IN503はモノクロ8階調，GT-3000はフルカラーで画像取り込みが可能だ。そして，Xファミリーならではの映像取り込みやテロップ機能もある。すでにX68000専用に発売されている周辺機器カラーイメージユニットを利用すれば32768色，256色，16色，モロクロによるスチルでの映像取り込みが簡単に行える。

そしてこのオプションでは，あのアンドゥ機能があるのだ。うわっと，失敗！というときに，いまのはなかったことにしてくれるたいへんありがたい機能である。このほか，縦2画面のレイアウト表示や，縦横任意の比率の方眼を引く機能などもある。

PROはいかにして色を選ぶか

X68000が一度に表示できる色数は最大65536色。これは，R.G.B.にそれぞれ5ビットずつ，共通の輝度として1ビットが割り当てられることによる。ただこのZ'sSTAFF PRO 68Kでは最大32768色表示（おそらく共通輝度の1ビットがマスキング情報に与えられるのだろう注2）となっているようだ。えっ，半分しか出ないの？　なんて思うかもしれないけど，これだってものすごい色数である。赤だけで32階調もの色が得られることになるのだ。これだけの色があると，色の選び方からして問題となる。

パレットのウィンドウを開いてみよう。お馴染みのスクロールバーがついたカラーサンプルが見える。32色しか見えていないが実際には256色のカラーをストックして

注1

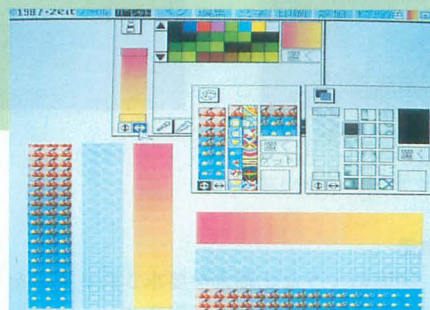
X1turboもZになって4096色となり，機能的にも環境的にも8ビット機の中ではもっとも充実したものとなっている。Zの隠れ機能は2月号で瀧山氏が紹介してくれた。FM音源については祝氏がMMLを発表している。しかし私の期待は満たされていない。それはX1turboZに標準でついてくるZ'sSTAFF-Zがサブセット版だからである。おまけとはいえ，8色しかないX1turbo用のZ'sSTAFFのほうが機能が高いというのは困ったものである。これはメーカーの責任でもあるはずだ。

注2

もちろん，65536色出ればそれに越したことはない。しかし，マスキング情報がデータとして保存できるなら，そのほうがずっと価値があることだ。考えてみてほしい。日本のメーカーが65536色とか26万色とかいっているのは，使い方をまったく考えないで，色だけ出せばこれだけ出せるという上限の数だ。ちなみにゲームセンターのスペースハリアーの機械だって65536色分のビットをもっているが，1ビットは透明色に割り当てられているらしい（重ね合わせ処理のため）。

左上）パレットのウィンドウは究極の絵の具箱だ。円形のカラーチャートは描きたい絵の色調に合わせてS（彩度），V（明度）を設定しておくとよい。

左下）タイル，トーン，ペンをエディットするウィンドウだ。◀▶のマークをクリックすると全体の階調が1段ずつ変化する。ちなみに，ペンは4×4，8×8，16×16の3種類の太さが用意されている。

右下）パレット，タイル，トーンによるグラデーションだ。このすごさ，わかります？

おける。中央の白い部分は色を混ぜるためのスペースだ。つまりこれは正しくパレットであり，絵の具を混ぜ合わせて自由に好みの色を作り出すことができるのだ。ぐちゃぐちゃに重ね塗りした色の上をマウスで左クリックしながらなぞるようにすると，スポイト型のマウスカーソルが指しているドットの色がパレット右上に表示されるため，気に入った色になったところで指を離せばその色が選択される。必要ならカラーサンプルとして登録しておけばよい。やってみると本物のパレットに比べてはるかに簡単でしかも快適だ。いきなり感心してしまう。もっともこれくらいで驚いていては先へ進めない。

色を選択するにはほかにも方法がある。画面の任意の場所からスポイトで色を取ってくることももちろんできるし，美術の時間に習う色相，彩度，明度による指定もできる。HSVのところをクリックすると円形のカラーチャートのようなものが現れる。これは角度によって色相が変化し，中心に向かうほど明度が下がるようになっている。また彩度は，その下のSのところを操作するとチャート全体の彩度が変化し，Vのところを操作すると明度の範囲が変えられる。これもマウスでもってひょーいひょいである。メリハリのきいたポップな絵を描きたいとか，全体に淡いパステルカラーで，といった色調を重視する場合には彩度や明度の範囲を固定してチャートから選ぶとよいだろう。

また，チャートから色を選ぶと右側のHSVの値を表すバーは自動的に変化するし，逆にこのバーを直接マウスでコントロールすることもできる。いま選んでいる色よりもうすこし赤みのある色をとか，若干控えめな色を，あるいはほんのちょっとだけ明るい色を，といった雰囲気で選びたいときには有効だ。

さて，ここまでに紹介したのは，いわば色を選択するための基礎となる機能である。いちいち色を混ぜたりHSVで作ってもかまわないのだが，実際に絵を描くときにはもっと手早くやりたいものだ。使いそうな色はあらかじめ作ってカラーサンプルとして取っておくのもよい。セーブすることもできるから，作業を中断するときや同系色の絵を描くときにも便利である。

PROはいかなる表現も可能だ

色は選ぶだけではどうしようもない。いくら何万色も使えるからといっていちいち色を変えていたら一生かかっても絵は完成しないだろう。このZ'sSTAFF PRO 68Kでは筆のタッチで色の濃度が変えられたり重ね塗りによって色合いが微妙に変化する。たとえば周囲のにじんだ線を描くことのできるペン，地の色と混ざり合う透明水彩画のような表現ができるペン，が利用できるのだ。

パソコンのグラフィックが多色化すれば当然こんな機能もできるだろうと密かに期待はしていたが，実際に自分の手で試したときの驚きはまったく新鮮なものである。しかもこれは実に気持ちがよい。ペンや色を変えながらついつい無意味な試し塗りを続けてしまう。もしかしてこの快感が当たり前のようにならなくてはまともな絵なんて描けないんじゃないかと思ったりするほどだ。

PROはエディットの鬼だった

とにかくなんでもかんでもエディット可能にしてしまうのがZ'sSTAFFのいいところだが，いままで8色しか使えなかった時代には，ペンにしろスクリーントーンにしろ形さえ決めれば，まあそれ以上はどうしようもなかった。しかし32768色ともなると恐ろしいことができものだ。ペンやスクリーントーンのエディットは方眼のマス目を単に黒くするのではなく，なんと32階調のトーンで塗り分けることができる。つまり，あの気持ちのいいペンが自由に作り出せるのだ。そしてスクリーントーン。アミ点の階調を変えることでいくらでも変化をつけられる。

さて，32階調のエディットを行うと，同じパターンでも，もう少し淡いトーンや濃いトーンといったものが欲しくなってくる。もちろんいちいちエディットし直したりするのは大変。そこで便利なのがペンやスクリーントーン全体の濃度を1階調ずつ上げたり下げたりする機能だ。これもまた極楽であろう。

お馴染みのカラータイルも紹介しないわけにはいくまい。Z'sSTAFF PRO 68Kの進化した機能を見るとあの栄光のタイルパターンでさえも色褪せて見える。しかし色褪せた表現も大事だ。いままでの8色だけのタイルではどうしてもギラギラとした感じになってしまったが，中間色を利用したタイルなら甘さも優しさも表現できる。古びた石畳を敷いて枯淡の境地に至るのもよい。

なお，スクリーントーンとタイルは4つのパターンを組み合わせて32×32ドットの大きなパターンを作ることもできる。実をいうと，さらに怖い機能があるのだが，それは次の項目で触れることにする。

PROはグラデーションの達人だ

便利といえば極め付けなのがZ'sSTAFFの名物ともいえるグラデーション機能であろう。任意の2色の色の間で連続的に変化する色を作り出す機能のことだ。これさえあれば似たような色をいくつも作る必要はまったくない。と，ここまでは従来のZ'sSTAFFのお話。PROは違う。

Z'sSTAFF PRO 68Kのグラデーションは色を選ぶための補助機能としてだけあるのではない。Z'sSTAFF PRO 68Kでは，

作ったグラデーションパターンを画面上のいかなる領域に対しても表現することができるのだ。これがZ'sSTAFF PRO 68Kの(ちょっとしつこいな)グラデーション効果というものなのである。

グラデーションは縦方向と横方向のいずれかを選択することができ，ボックスフルでも領域内ペイントでも好きな機能と組み合わせて使う。なんて簡単なんだろう。グラデのかかった空なんて一瞬じゃないか。どんどん使ってみたくなる。とうぶんの間私の描く絵はグラデーションのかかったものばかりになりそうだ。

グラデーションの攻撃はまだやまない。怖い機能とはこのことだ。まずは任意のタイルの間でのグラデーションができる。これも色の場合とまったく同じで，画面上にワンタッチでタイルパターンによるグラデーションが表現できる。色違いの同じパターンを掛け合わせれば幻想的な気分に浸れること請け合いだ。そしてスクリーントーンまでもが！　いったい私はどうすればいいんだ。

マスキングの例をひとつ。マスキング，マスキング消しゴム(?)，反転，オールクリアの4つの機能を合わせて使用する。マスキングされた部分は青く点滅する。そのままセーブすることも可能だ。

PROはおやじをマスクする

これは来月に回そうかと思ったのだが，もはや私はすっかり愛用してしまっているので簡単に紹介しておくことにする。

マスキングというのは，おもにエアブラシなどを使う際に，絵の具やインクが余計なところにつかないようフィルムやシートで特定の部分を覆い隠すことをいう。人物を一時的に隠して背景を塗り変えるといったときには非常に便利である。また，上から絵を重ね描きしたいときにも有効だ。

たとえば背景に重ねて鬼太郎のおやじの絵でも描くとしよう。せっかく描いた背景を汚さずに鬼太郎のおやじを描くにはどうすればよいか。鬼太郎のおやじだけ描いた別のファイルを用意して合成すればよいかもしれない。だけど，なにも鬼太郎のおやじのためにわざわざ別のファイルなんか作りたくない人だっているだろう。そんなときにマスキングを使えばよい。

まず鬼太郎のおやじのシルエットをマスキング機能を使って描く。はみ出してもマスキング解除機能があるから安心だ。いずれもすべての描画機能と対応しているからかなり自由に描ける。出来上がったらマスキング反転機能を使用する。これで鬼太郎のおやじは君の思いのままだ。納得がいくまで描き込んだらマスキングをオール解除にすればよい。

PROはマルチウィンドウである

作画中には，必要に応じてさまざまなウィンドウを開き，目的の機能を引っ張り出して使うわけだが，状況によってはいくつかのウィンドウ間を頻繁に行き来しなければならないことも多い。turboZ'sSTAFFを使っていたときでも，パレットとペンとルーペ機能のウィンドウを多用するため，一度に開けるウィンドウがひとつだけなのはやや辛い気がしたものである。ところがこれはX68000である。Z'sSTAFF PRO 68Kではいくつでもウィンドウが開けるのである。これは便利といわざるを得ない。

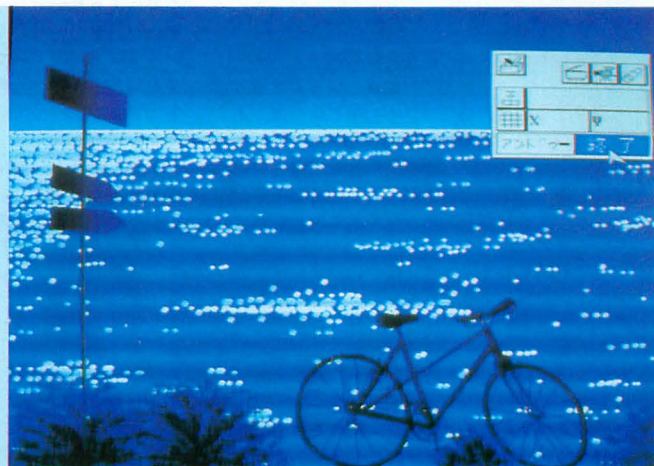
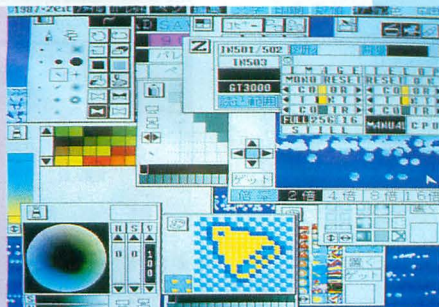
エディット&エフェクト(の予告)

turboZ'sSTAFFが出たとき，表現力の素晴しさに比べると編集機能には若干のパワー不足を感じたものだ。とはいうもののメモリの少ないX1turboにあれ以上を望むのは酷である。メインメモリが64Kバイトしかない機種でアンドゥがないなんて文句をいわれたら五代くんだって黙っていない。しかし，X68000ともなると話は違う。というわけで次回はイメージエディットの話を中心にZ'sSTAFF PRO 68Kの化け物ぶりを追ってみよう。

おまけ

Z'sSTAFF PRO 68Kの価格は58,000円の予定となっている。「え〜っっっっっ！　待ってください。響子さ〜ん，捨てないで〜っ！」と夢にうなされる読者もいるに違いない。「貧乏って悲しいな」そんな声も聞こえてくる。無理もない話だ。実際にX68000を買ったばかりでお金のない人には気の毒である。しかし，私は決して高すぎるとは思わないのである。べつに花子が58,000円で同じだからというわけではない（花子はぜんぜん違う種類のソフトだ)。Z'sSTAFF PRO 68Kは従来のグラフィックツールを超えたグラフィックツールであり，パーソナルコンピュータのソフトウェアとしてはほかに比較できるものがないからだ。

X68000用　5D(2HD)版　58,000円
ツァイト　☎03(342)4669

次回をお楽しみに！

ウィンドウはいくらでもオープンできる。しかしこうなるともう絵は描けないな。

SOFTWARE INFORMATION

話題のソフトウェア

速報！　X68000版スペースハリアー

まずは右の写真を見てください。もうコメントなんかいらないかもね。これがゼビウスを発表してくれたばかりの電波新聞社さんの秘密兵器，X68000版“スペースハリアー”です。このゲームは大阪で開かれた「マイコンショウ'87」にも出品されたらしいので，もうご存じの方も多いのかもしれません。編集室に届いたサンプル版を見たら，これはもうただただ驚くばかり。周囲から「おぉ！」との歓声が上がったあとは，絶賛の嵐。よくぞここまでと感謝，感謝の大サービス。

このスペースハリアーでは，敵の弾以外はほとんどスプライトは使っていないようだけど，とにかく速い，キレイの言葉だけで十分みたい。まだ音楽もないしキャラクターも全部揃ってはいない未完成のサンプル版でも，十分過ぎるぐらい楽しめちゃう。これぐらい凄いできのゲームがボコボコ出てきて，こんなのがX68000のゲームの水準となってくれればあとはもうしめたもの。「次に出てくるのはなにかいな」とジッと待っているだけの毎日が楽しくなってしまいそう。ほんとにこの完成版はどうなるんでしょうね。このサンプル版よりまだ速くなってしまうと，きっと手に力が入り過ぎちゃって，ジョイスティックが何本あっても足りなくなりそう。そのうちスペハリ専用“超合金ジョイスティック”なんてのも発売されちゃったりして。

どうでしょう，写真だけでもこの迫力。誰ですか，いまからX68000用のムービングシートまで期待しちゃっている人は

アクションゲームでさんざんはしゃいだあとは，ジックリ楽しむRPGのお話。7月18日発売予定で現在移植が進行している，**“ウルマイマⅣ”**のX1版の画面写真をご紹介しましょう。右下をご覧ください。これがその写真です。なかなかのものでしょう。このゲーム，予定どおりならばこのOh!MZ 8月号の発売と同時に店頭に並んでいるはずですが，いかがなものでしょうか。X1/X68000のゲームファンにとってはこの2本，今年の夏の強烈なお中元となりそうですね。

ゲームソフト発売日速報

先月，このコーナーでお知らせしたばかりのリバイバーの完成版がアルシスソフトさんから編集室に到着しました。発売は間違いなく7月中旬とのことです。このリバイバーは

これが待望のウルティマⅣ X1版です

こうしてキャラクターが設定されていくのです

読者が選ぶ今月のゲームベスト10

こんにちは皆さん，毎度お馴染み，今月の人気ゲームトップテンです。盛夏ということばから連想するのは，照りつける太陽と碧い海，あるいは頂に白い雲のかかる山。うーん，いいなあ。というわけですっかり夏の休日に飛んでいます。暑さをふっ飛ばすゲームといったらやはりシューティングアクションですよね。前方にビッグコア出現！ バキュンバキューン。おっとこれじゃ西部劇だ。やはり弾やレーザーはシュンシュンと飛び交わなくちゃ。でも私は反射神経には見放されているらしく，ラムちゃんのいとこのテンちゃんみたいにじたばたしてるもんだからいつもあえなく敗退します。誰か上達法教えてよ。

話は変わりますが，数年前，初めてパソコンでやったゲームは作者不詳のパックマンもどきでした。キャラクターが大文字と小文字のCで，交互に入れ替わりながら移動するようすが，口を開けたり閉じたりしてるみたいでかわいかったのを覚えています。文字にはとても表情がある。先日編集室を席捲したRogueなども，プレイしてるとアルファベットや記号がとても生き生きしてくるんです。

それにしても，学生の皆さんには長い夏休みがあってうらやましい。思いきり汗を流して遊んだら，編集室へのお便りも忘れずに。

1	三国志
2	大戦略X1
3	ディーヴァ
4	信長の野望　全・国・版
5	グラディウス
6	ウィザードリィ
7	ザナドゥシナリオII
8	夢幻戦士ヴァリス
9	殺人倶楽部
10	A列車で行こう

リバイバー

ホテルウォーズ

ガイアの紋章

ゲーム画面が従来のものとまったく違っているので，最初はとまどってしまいますが，こういったAGVの新派が登場してくれると，これまで謎解きや犯人捜しばかりに熱中して楽しんでいたAGVに，画面を見る楽しさも与えてくれてよりいっそう楽しみが増えてしまいそう。ゲームの内容紹介はできれば来月ご紹介したいと思っているのでお楽しみに。

ホテルウォーズ（7月中旬発売予定）

ヨーロッパを舞台にした思考型ビジネスシミュレーションゲームです。ヨーロッパのホテルには古くから伝統のあるものが多く，世界的なリゾート地も集まっているので，ほかのホテルと競合しながら経営戦略を巡らせるには絶好の条件を備えている場所でしょう。株の売買や他社の乗っ取りなど，複雑に絡んでくる裏取引をうまく処理して，ヨーロッパのホテル王を目指せ。

X1/X1turbo用　5D版2枚組 7,500円
ボーステック　☎03(407)4191

新作ソフトウェア情報

☆……7月2日現在発売中　★……近日発売予定

☆ガイアの紋章

ファンタジーシミュレーションゲームとして発売された「エルスリード」の続編の登場だ。ガイアに4つの国が栄えていた時代，ヴェルゼリアの王ボーゼルⅠ世は“闇の術”の使い手ベルナリッヒから術を体得したそのとき，ガイア統一に向けて侵攻を開始した。一方，“光の術”を光の魔術師バーリンより授かったエルスリードの王国ジークヘルマンは，ボーゼルⅠ世の行く手を阻止せんと立ち上がった。そうして一度はエルスリード軍に破れ去ったヴェルゼリア軍だが，ボーゼルⅠ世はまだ野望を完全に捨ててはいなかった。こうしてまたガイア大陸に戦火が広がろうとしている。このガイアの紋章ではエルスリードよりさらに魔法や戦術，兵器などが追加され，前回以上に緻密な作戦行動が楽しめるようになっている。

X1/X1turbo用　5D版2枚組 7,800円
日本コンピュータシステム　☎03(486)6311

☆ザ・キングサーモン

ディスプレイ上で釣果を競えるMZ-2500用釣りシミュレーションゲームだ。このゲームではキーナイ河でのトローリングとルアー釣り，SKILAK LAKEでのパイク釣り，バンクーバー島近海でのトローリングと4つの種類の釣りが，実際の釣りと同じような状況にもとづいて楽しめる。タナの取り方やラインの選択などが比較的現実のデータに忠実に再現されているため，釣りファンのマニュアルとしても役立つかも。また釣り上げた魚をプリントアウトできるオマケが付いているのには笑えてしまう。

MZ-2500用　3.5D版 7,800円
ホット・ビィ　☎03(360)3623

☆大戦略マップコレクション

ストラテジックゲームの代表作「大戦略X1」の登録ユーザーから公募した，オリジナルマップデータ16枚をセレクションしたソフトが登場した。このソフトは大戦略ファンにいま以上にゲームを楽しんでもらおうと企画され，今回特別にX1/88兼用版5,000本の限定発売を行うもので，限定発売の特典として大戦略オリジナル・カレンダーカードが付いているのも魅力だ。

ザ・キングサーモン

X1/X1turbo，PC-8801兼用　5D版 3,500円
システムソフト　☎092(521)0337

永遠のゼビウス

なにはともあれX68000用ソフトの第1弾である。「本物そっくり」といわれる移植版だが，やはり若干の違いが見られる。いくつかの点では本物を越えてしまっているといえよう。かつて完璧といわれたMZ-2500版もX68000版と比べると見る影もない。

まずはディスクをAドライブに入れて起動。このままジョイスティックで始めてもよいが，縁起物だから一度OPT.2キーを押す。懐かしのゼビウスが甦る。背景はマイコンショウでやっていたデモ版よりも美しくなったが空中キャラクターのギラギラした感じは再現されておらず少し残念。その代わり動きの滑らかさは絶品といっていい。

ゼビウス軍の攻撃は熾烈をきわめる。オリジナルでは編隊を組んで現れ3発の弾を吐いて優雅に飛び去る「カピ」は1機300点と稼ぎどころだったのだが，このX68000版ではどうやら自分を「テラジ」だと思い込んでいるらしく地獄の連射を披露してくれる。さらに通常モードでも十分難しいのだがスーパーモードとなると冗談ではすまないくらいの猛攻である。

また，ピピピピッと大気を裂いて黒玉の一群が空間転移してくる。反射的に迎撃に向かうのだが，よく見ると真ん中が赤く点滅してたりするのである。突然画面は花火大会と化し，プレイヤーは地獄に落ちる。場合によっては画面に100発くらいの弾が乱れ飛ぶのだ。あり余るCPUパワーである（しかし，限度というものも……）。

スーパーモードではちゃんとファントムやギャラクシアンも登場してくるようだし，起動時にマウスボタンを押すとなかなかかわいい反応を見せてくれる。コンティニューは……ふむふむ，面選択は……なるほど。伝統のパターンを踏襲しているようだ。この分なら無敵モードもあるかもしれない。VSを立ち上げるとアイコンも定義してあるのだが，なぜかコマンドモードで起動してしまう。うーむ。

いまさらゼビウス，たかがゼビウス，されどゼビウス。かつてX1を「ゼビウスのできるマシン」として世に知らしめ，ファミコンの売り上げを伸ばし，MZ-2500の底力を見せつけ，いまなおゲーセンのディスプレイにタイトルを焼き付けている伝説のゲームである。もちろんX68000はゼビウスくらいできて当然のハードウェアであるが，オリジナルのよさが素直に反映された移植だといえよう。もはや各面のすべてが懐かしの名場面，ゼビウスはやはり名作であった。

GR GAME REVIEW

今月はビデオゲームからの移植版「ドラゴンバスター」と巨大キャラつきアクションRPG「アルゴー」，そして超辛ロのアドベンチャーゲーム「マデリーン」を紹介する。「アルゴー」はアクションゲームとしてとらえるか，RPGとしてとらえるかで評価が分かれてしまったようだ。

ドラゴンバスター

ゲームセンターでお馴染みのファンタジックなアクションRPGだ。王女セリナを救うためドラゴンの山へ向かえ。

▼ひところゲームセンターでしきりに「バシュッ，バシュッ」と音をさせていたドラゴンバスターですが，どうにかX1にも移植されました。このゲームは最近雨後の筍のように乱立していたRPGもどきゲームとはある意味で一線を画しています。いい意味での開き直りとでもいうべきものがあり，ゲームバランスもうまくまとまっているようです。移植版だからといえば確かにそうですが，専用に作られたゲームより移植版

のほうがいいというのはやっぱり困ったもんだという気がします。

しかし，移植版の欠点というものもあるわけで，どうしても画面や操作性，速度が業務用のゲーム機にかなうはずがありません。このゲームには操作性に多少の難があるのですが，フィーリングと画面のどちらを優先するかというのは難しい問題なわけです。ドラゴンバスターの場合は，やはりあの独特の操作感がポイントですから，そこに重点をおくべきだったと思います。

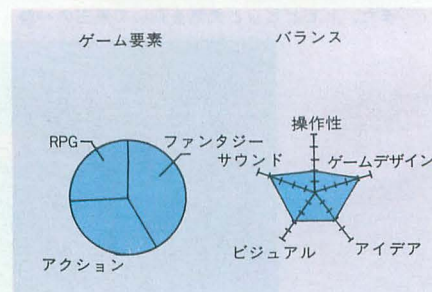
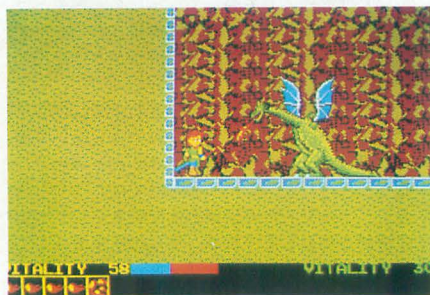
熱中度　▶▶▶▶▷▷▷　M.Y.

▼例によって例のごとくアーケードゲームからの移植版だ。RPG風の味付けはしてあるものの典型的な面クリア型のゲームである。移植版として生まれたからにはオリジナルと比較されるのは宿命だが，このゲームに関しては比較なんかしないほうがよい。それほど格差が大きいのである。このタイプのゲームはパソコンに移植するべきではないようにさえ思う。そんなわけで，ドラゴンバスターがX1でできる！　などとぬか喜びしてもいられまい。きれいな画面とか滑らかなスクロールといった「小さな」期待はことごとく裏切られることになるだろう。ソフトハウスのおにーさんは移植している暇があったらオリジナルをビシバシ作ってくれないものだろうか。

確かに知名度の高いアーケードゲームを移植したほうが営業面では得なんでしょうけどね。ところで，各面の最後にいるドラゴンの弱点がぜーんぶマニュアルに書いてあるのって，ちょっと軟弱すぎるんじゃないのかなぁ？

熱中度　▶▶▶▷▷▷▷　T.T.

X1/X1turbo用　5D版　6,200円
電波新聞社　☎03(445)6111

アルゴー

ギリシャ神話の世界をベースにしたRPGだ。画面は高速描画，戦闘シーンではリアルタイムゲームに早変わりする。

▼このゲームはずぇったいにマルい！　ストーリー展開は素直だし，グラフィックはよく描けているし，名物といわれるデカキャラも納得させてくれる。こと空中におけるデカキャラとの戦いは動きもよく，一度やったらヤミツキになってしまうぐらいの優れもの。320×200とはいえ，敵の投げる岩以外はすべてグラフィック画面に書き込んでいる点にも好感がもてる。X1のポテンシャルの高さを見事に証明してくれているといえよう。2年前ならこのソフトは親の遺言級と呼ばれたかもしれない。私にとっては“受験生のワードナ”とでもいうべきであろうか？

最近のゲームは「解かせないためのワナ」が多く用いられているケースがほとんどだが，そんなたわけたこともなく，初心者にも上級者にも自信をもってすすめられる（でも簡単には解けないよ。念のため）。

あなたもリアルワールドへということで，6つ星をさしあげたい。

熱中度　▶▶▶▶▶▶▷　S.K.

▼立ち上げて2分……嫌な予感がする。これはひとり旅版ブラックオニキスか？　5分たった……もしかしたらありとあらゆる扉を開けねばならないのだろうか。20分経過……悪しきRPGの伝統を潜やかに受け継いでいそうだ。いや，まだレベルは1だし最初の街さえ出ていないではないか。45分後……おっと，街を出たらいきなりデカキャラ登場だ。本当にデカい。3DRPGでデカキャラとは。しかもいきなりアクションゲーム。これはたまげたわい。

このゲームは，ギリシャ神話を題材にしたある王子の冒険RPGである。良くも悪くもRPGである。街や野原を歩きまわってア

イテムや情報を集めて，敵を殺してレベルを上げて，と基本的にはなんの変哲もない。少々速いだけだ。しかし，空飛ぶ靴で空は飛べるし（本当に空中を，雲の中を移動できるのである），壁はジャンプシューズで越えられる。デカキャラとの戦いもすごい。あとはバランスですな。

熱中度　▶▶▶▷▷▷▷　K.Y.

X1/X1turbo用　5D版2枚組　7,500円
呉ソフトウェア工房　☎0486(46)0660

マデリーン

きっかけは古道具屋で見つけた1枚の絵。伝説的な悪女マデリーン妃の真の姿を求めて過去を探る旅が始まった。

▼アイコンなどによるコマンド選択方式が主流の中にあって敢えてコマンド入力方式をとったゲームだ。アドベンチャーゲームの場合，コマンドがヒントになってしまう場合があるのでそれを嫌ってのことかもしれないが，やはり少々やりづらくなっている。いわゆる“アタッチソフト”なのである。コマンドがわからないことが原因で詰まったりすると，もうやる気がしなくなってしまうだろう。また，このゲームはマッピングができない。東西南北，加えて前後左右の相互関係がメチャクチャだ。そのため一度迷うとまず抜け出せないのではないだろうか。

このゲームのオープニング画面は目を見張る素晴しいものがあるのだが，そのセンスがゲーム内容に全然行き届いておらず，プレイ中のグラフィックははっきりいってダサイ。よかった第一印象がゲームを始めるとぶっ飛んでしまうのは残念でならない。

シナリオにはなかなかよいものがあるのだが。

熱中度　▶▶▷▷▷▷▷　H.K.

▼コマンド選択方式でないアドベンチャーゲームは言葉探しが主体となる。ものの名前がわからなければつまらぬところで立ち往生するはめになるが，メッセージに出てきた単語をかたっぱしから入力してみて，目当ての単語を発見したときはなぜか感激したりするものだ。父親を毒殺したといわれるマデリーン王女の背景にある真相に迫るという大きなシナリオの流れの下に，そういった小さな感激を積み重ねながらこのゲームも進んでいくわけであるが，最近のコマンド選択方式に慣れきった頭にはちょいとキツいかもしれない（塔の中で巻き物を手に入れる方法にすぐ気づく人は偉いと思う）。

まあ，最近軟弱なアドベンチャーが多くなったと嘆いている人にとって，格好のゲームになることは間違いないだろう。

ところで，副題になっている「亡き王女のためのパヴァーヌ」という文句を見ると，ラヴェル作曲の有名な曲を思い浮かべてしまうが，このゲームはそれとは無関係なようである。

熱中度　▶▶▶▷▷▷▷　A.N.

X1/X1turbo用　5D版2枚組　3,200円
ブラザー工業　☎052(263)5895

上海物語

ちかごろPC-9801あたりではやっているのがご多分にもれず「上海」だ。あちこちの98が上海マシンになったのはいうまでもない。そればかりか必殺エミュレーションソフトによって，2MバイトのRAMを増設し80287を積んだばかりのMZ-2861も上海マシンとなりはてた。

このソフトはトランプでいう神経衰弱やピラミッドに似たルールで画面上の麻雀牌をすべて取り除くという内容のゲームだが，シンプルなわりに一度手を出すとなかなかやめられなくなってしまう。操作は簡単，ルールは明快，ゲームは難解とハマるゲームの要件を満たしているようだ。

このゲームをやっていると，いつのまにか背後に人の気配がする。そして矢継ぎ早に指示を連発され，プレイヤーは背後の声に操られるオペレータと化してしまうのだ。当然，結果はあまりよくならない。「馬鹿め，死にゆく牌をつかんだな」，ほとんど『風の雀吾』の世界だ。

なかにはX68000で上海モドキ，「香港」を作ろうとしている者もいる。基本的なゲーム部分は単純なのだが，どうやらネックになっているのは牌のデータらしい。麻雀牌ではそのものだから，トランプにしようとか配置を変えたほうがいいとか。そのうち65536色版香港とかが現れるかもしれない。

ゲームグルメ集まれ

Shimizu Kazuto
清水　和人

出番を忘れて舞台裏でジッと隠れていたキャラクターに，急にスポットが当たるとどうなるか。お粗末な大根役者の演技か，はたまた目を見張るような名演技か。このゲームの場合は幸いにも後者であった。それも遊ばせる心を忘れないままの。

X1/X1turbo用 5D版2枚組 7,800円
テクノソフト　☎0956(33)5555

ウィークデー午後6時半NHK。健全な家庭のお茶の間を飾っていた，ズンチャッチャンチャカチャッチャッチャッのテーマソング。仁義礼智忠信孝悌，とここまでくればクイズ番組では，真っ先にマッハ文朱(小林亜星もあり得る)のランプがつく。滝沢馬琴の「南総里見八犬伝」に原作を置く人形劇「八犬伝」。当時誰もが「我こそは玉ずさがおおんりょうなり」と見栄を切り，八犬士の名を覚えたものだ。

それは飛び散った8つの玉を探すというテーマだった。つまりこの「九玉伝」を見るとき常に思い出すのが「八犬伝」であり，「仁義礼智忠信孝悌」とひとつずつ文字の彫られた玉なのである。「九玉伝」では玉に文字はなく，八犬士も登場しないが，昔からあるロールプレイのパターンの中で，時代劇を感じさせる画面と9つの玉が私のこのゲームに対する開始時刻を決めたといえよう。このゲームの開始時刻は，常にウィークデーの午後6時半，場所は茶の間に限る(「目黒のさんま」的だなあ)。

このゲームの質

今回はプレイの内容に入る前にこのゲームの質についてふれてみたい。つまり「このゲームはなんなのか，いいのか悪いのか」である。

結論からいってしまうと私個人の見解としては，「このゲームはいい」のである。最近ゲームに対する過度の期待がはびこっているようだが，このゲームのよさは「いかにもゲームらしい」という保守的なよさなのである。革新の売りものは「新しいこと」だが保守の売りものは「洗練」である。このゲームはじつに自然になじめる質を持っている。

ゲームを分類することはあまり好まないが，プレイヤーをよってたかって痛めつけるゲームと，プレイヤーを適当に遊ばせながらしかし簡単には進ませないよ，というゲームがある。このゲームは後者であり，それは最近のロールプレイの常道でもあった。しかしこの「遊ばせる」，「簡単には進ませない」という2大テーマをいかにうまいバランスで実現するか，についてはなかなか優れたゲームがなかったのである。いまのゲーマーはぜいたくで，「進めない」ゲームは飽き，「遊ぶだけ」のゲームは簡単すぎるといってこれも飽きちゃうのである。このような「ゲームグルメ」全盛の時代に出た「九玉伝」を，私はこれら2大テーマに対するひとつの解答と見てよいのではないかと思う(フー疲れた，「ゲーム内ゲーム論」だ)。

これはちんねん，そんねんの2人モードです

いろいろ教えてくれる易者のおじさん

八方数珠が使えればこっちのもの

やたらとお金のかかる白浜海岸

村

ゲームが始まるとちょっとおかしいところが出てくる。セーブのための名前を決めるのだが，KAZUTOと入力したあと，ふと不思議な気持ちになった。なにせひとりプレイの場合は「ちんねん」という若僧に決まっているからである。まあそんなことはいい。

まず1面は「寿村」という寂しい村である。最初は周りの様子を見てる暇もなく敵キャラが寄ってくるのだが，まあそれでも1，2発ではやられないので，バシバシ数珠攻撃をしながらうろついてみる(ちなみにこのゲームは2トリガーのジョイスティックを使用するのがよい)。森に囲まれたその村は墓地や井戸などがあるほかはなんの変哲もない昔風の村である。

まず最初にするべきことは，アイテムをがさごそと集めることである。必要不可欠なのはろうそく，ちょうちん，である。集め方はカンタン。敵をやっつけて出てきたアイテムを取ればよい。攻撃は投げ数珠一本でよいだろう。森沿いに進みながら撃ちまくれば敵のほうから当たってくれる。気楽にアイテムをかせいで安心感を得る，これがこのゲームの「遊ばせる精神」である。

中でも嬉しいアイテムはまず赤い笠である。これをかぶると無敵になり，黄色から白へと変色してなくなるまでその効力は続

く。もうひとつは八方数珠，これがあると凄い。ほとんど半永久的に敵をやっつけ回し，アイテムをごっそり集めることができる。赤い笠と八方数珠を身に付けたら，白数珠・赤数珠（八方が消え，１発２発になってしまう）さえ取らなければ，無敵も同様である。とはいえ無敵自体に大した意味がない。これがこのゲームの特徴である。

さてこのゲームでのヒントは店というところで聞ける。店は地下や家の中などさまざまなところにあって探さねばならない。ここでノウハウをひとつ。「ひとつの店に４回入れ」である。ひとつの店は４回入るとそれぞれ「ヒント」「武器」「食料」「タダでなにかくれる」というように内容が変わるのだ。これが悪魔の支配する世界の恐ろしさである。このうち武器と食料は有料だが，「ヒント」，「なにかくれる」は無料なので，おおいに利用するとよい。

もうひとつ地下への穴や家を発見したら入ってみることが大切だが，入る位置が少しずつ違うことがあるので注意しよう。また地下への入り口は，木・岩・井戸・墓などで巧みに隠されているので発見するのに苦労するが，それだけに新しい穴が見つかったときはとてもエキサイトする。「やりーっ」。この瞬間こそがこのゲームの「喜ばせ方」である。じつに適度な隠し方である。

さて店でのヒントを基に，ある井戸のところで××すると２面へ行けるのだが，ち，ちょっと待ってほしい。忘れてはいけないのはこのゲームの大テーマ「玉を探す」である。それには各面に１匹ずつ隠れている怪物の親玉を倒すことが必要だ。命をためておいて（面×100？）一気に数珠攻撃をかける。

まあやられそうなら一度退いてもよいが，まずは１回で仕止めたい。「やればできる」という根性があればたぶん大丈夫，何度もいうようにこのゲームのテーマは「よってたかってゲーマーをいじめる」ことではなく「遊ばせる」ことにある。謎解きに重点を置いているのだ。怪物に負けるようでは鍛え方が足りないといえよう。かくて２面である。

山

２面はいかにもダンジョン風で始まる「悪霊山」である。最初は暗闇から始まるがここまで来たからは，ちょうちんぐらいたくさん持っているだろう。一つ目小僧が攻めてくるがどうってこたあない。むしろ手強いのは九尾のきつね，こいつあ玉を撃ってきて八方数珠を取り上げてしまう。まあそれでも別にかまわない。

例によって店に４回ずつ入っていく。ヒントが出たりお金が出たり，ぜいたくのしほうだいである。それでも時々アイテムをちょうちんにセットしたまま撃ちまくって手間のかかるアイテム集めをしなければならないこともある。やってしまったあと，「あ～あ，でもまあいいや」と立ち上げ直さずにプレイを続行するのがこのゲームのよい点である（でもセーブしたほうがいいよ）。この面はろうそくさえ正しく使えれば軽くクリアである。さて，次！

原

村，山ときて次は原である。「悪霊山」から「みどりが原」とまた平和そうな原野に出た。相変わらず店は４回，先のほうの面のヒントもあるからメモしておくこと（そういえばこのゲーム，メモはいるがマップ作りは必要ない）。

この面は火炎鳥が支配している。このへんになると，倒すのに時間がかかるが，要は頭部を狙って撃ちまくるのみである。むしろ地下への入り口を探す際に，怪しいところがやけにいっぱいあるのでこっちのほうが大変である。

よく考えると，２～３種類の店と怪物と，次のシーンへの入り口を探せばよいのだが，３面ぐらいから，重要な穴は見つけにくくなっている。しょうがないから敵をやっつけてうさ晴らしをしながら進撃するわけである。

峠

第４面は大雪峠，一面の銀世界，いや白世界である。なかなか風流であり，このゲームのデザインセンスを楽しみながらプレイを進める。ただ，数珠やアイテムがちょっと見にくいのだが，一撃必殺のアクションゲームではないので別にかまわないだろう。

この面は地下もかなり大変で，穴に入ったらいきなり２面へ戻ってしまったりする。そんなときはセーブが役に立つが，暇ならもう１回せっせとやり直すのもよい。アイテムもそのほうが楽に手に入ることだろう。なにごとも経験である。

さてこの面の怪物は予想どおりの雪女であるが，しばらく知らん顔していて突然攻撃してくるところがなんともかわいい限りである。例によって勝てるけれども，たまたま命が少なかった人は途中で薬のツボを使用するのもよいだろう。そしてかなり歩きまわらないと次の面に進めないのが，ここである。

町

お次は「椿城下町」忍者のウヨウヨいるいやらしい面である。しかしさらにいやらしいのは怪物のいる場所がなかなか見つからない点である。私などは何度もリセットしてしまってからようやく気がついたくらいのいやらしさである。地下への入り口が隠れているのである。

ところでこの面では門を通ったりするがそこではアイテムの鍵が活躍する（なーんだ，ここで使うんだったのか）。そして無限地獄にはまったかと思うようなところに入ってしまう。だからリセットした。これが私の犯したミスである。無限地獄にも１点のゆるみがあったのだ。それは他の面で岩などを撃っているときと同様，数珠が飛ばないところ，つまりなにかが数珠をさえぎっている場所である。それさえ見つかればこの面は終えたも同然，しかしワナにかかると１面行きである。せっかく到達した５面からアッというまに１面。しばしボウ然とするがそれも人生である。

岸

そして６面は「白浜海岸」。いきなり狭い島に来てしまうが，実は地下を通じて他の島とつながっている。しかも渡るごとに60も金をとられるのがいたい。帰るときも60とられるので，人によってはいったん前の面に戻ってまた進んでくるのもひとつの方法かもしれない。

さてこの島では地下への入り口はさらに探しにくくなっており，地面に「DEVIL」の文字が彫ってあったりして，わけがわからんのだ。ろうそくを随分使ってしまったし，岩は１回１回押してみるしで，かなり苦しい戦いである。合間に敵をやっつけるくらいが楽しみだが，アイテムもいいかげん満杯で，あとは穴探しだけが楽しみとなってしまった。

というところで，うーむあとは「お・ま・か・せ♡」である。遊ばせてくれるソフトゆえに，決して無理な謎はないハズである。楽しんでいるうちにいつしか解けてゆく，この謎解きのプロセスをじっくり味わってほしい。

君はブライアンになれるか

Yoshida Kouichi
吉田 幸一

特殊研究施設を占拠したバイオドールをせん滅するため，ブライアン中尉はガレオスに搭乗し研究所内に潜入する。コクピットのスクリーンが映し出す映像に，しだいにプレイヤー自身の感情が交錯した世界にのめり込む，そんなSFアドベンチャーだ。

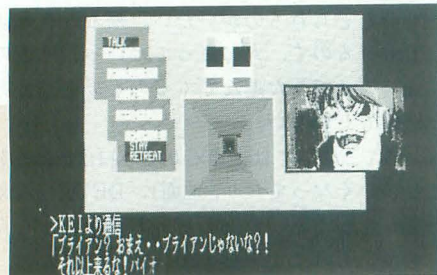
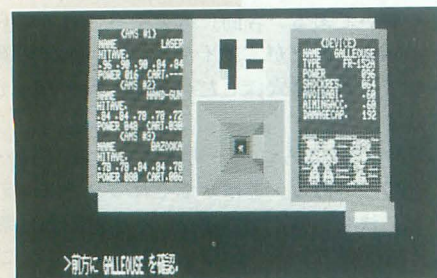
あなたはマクロスが好きか。古い？ では，レイズナーは好きか。ドラグナーはどうだ。私もかつて例にもれずガンダムやボトムズを好んで観たものだった。パワードスーツ，モビルスーツ，装甲機兵，ひいては戦闘メカアニメは好きか。よろしい。

カーマイン X1は，そんなあなたを満足させるSFアニメ風ストーリー展開とビジュアル効果を併せ持ったSFアドベンチャーゲームなのだ。

時の涙は見れなくとも，人形の涙くらいは拝めるかもしれない。

AFWとはなんぞや，と問えば

カーマイン X1にはユーザーズマニュアルのほかに，ゲームやアニメに欠かせない設定資料集のような小冊子とマップが同梱されている。前者には敵・味方のメカ・キャラのデータや今回の作戦指令書などがマニアックなイラストとそれっぽい文体に彩られてばらまかれていて意外と有用である。

さらに有用で役立つのが，“特殊研究施設カーディナル”のマップ（見取り図）である。このゲームで直接の舞台となるのはマップに示されているカーディナル内のみだから，面倒なマッピングはいらない。

疑心暗鬼にかられるキース

ガレオスを確認。味方だ

肝心のストーリーに目を向けるわけだが，その前にこの手の物語，アニメ・ゲームにつきものの特殊名詞，キーワードについて言及せねばなるまい。AFW，バイオドール，メカドール。

AFWはアーマード・ファイティング・ウォーカーの略である。なんとまあかわいいネーミングだが，モビルスーツ，ウォーカーマシン，メタルアーマーなどなどロボットアニメでは戦闘メカの呼び方に苦心してきたわけで，AFWもその轍を踏んでいる。このAFWのひとつ“ガレオス”にプレイヤー（ブライアン）は搭乗し，任務を果たそうと奮闘するのだ。その全高は4.61メートルとかなり小さい。昔は「しーんちょー57めーとる～」などと大きさを誇っていたものだが，ガンダム以降急速に戦闘メカは縮んでいるのだ。これも時代の流れか。

続いてドールである。“陸軍で開発中の小型陸戦兵器”だそうで，ヒューマノイド型である。メカドールとバイオドールがあり，名前からその違いはわかるだろう。10年前のSFならアンドロイドやロボットで片付けられていたものだが，またもやネーミングで苦心しているところが泣ける。

では，作戦目的である。

1) 内部に残されたターゲットの破壊処分
2) 研究所内中央管理コンピュータの制御権奪回
3) 残存生存者の確認および救出

つまり，カーディナルで開発中のバイオドールが反乱を起こし施設を占拠したため，潜入しすべてのバイオドール(ターゲット)を破壊し，まだ生きている研究員がいたら救出しろということだ。

私は吉田幸一だ

アレード班がAエレベーターから，マーコス班がBエレベーターから，4体のメカドールがCエレベーターから侵入して作戦は始まる。あなたはアレード班のブライアン・W・カーニハン中尉である。私は吉田幸一である。しかし，プレイヤーはなにがなんでもブライアンなのである。性格もブライアンなのである。女の子を見たら助け上げ，その子がふさぎ込んでいたら慰めてやる，任務に忠実なブライアン中尉である。「私はブライアンだったんだ」と思うことができればこのゲームはひときわ楽しいものとなるだろう。急げブライアン！

ガレオスは2人乗りだが，ブライアンの後部座席は空である。このあたりのことはマニュアルに詳しい。ゲームスタート時にはすでに作戦は始まっており，それぞれ担当区域へと散ったあとである。あとは任務遂行を目指して基地内を探索するのだ。

こういった前置きは，日本テレネットなら（ヴァリスやアルバトロスやファイナルゾーンのソフトハウスですよ）長々とアニメにしただろうなあ，幸か不幸か。

では，ガレオスのコクピットへとブライアンになったつもりで移ろう。

中央にメインスクリーンがあり，ガレオスのTVカメラが捉えた映像が映し出される。その上にはマップがあり，現在地点を中心に5×5の範囲が表示される。

画面左がメニュースクリーン，右がステータススクリーンである。通常は選択できるコマンド，自機のステータスが示されているが，状況に応じてクローズアップ画像や通信映像なども表示され臨機応変に盛り上げてくれる。画面下4分の1がメッセージスクリーンであり，ガレオスからのメッセージ（“>”に続くメッセージ）やブライアンとほかの人との会話（なんとプレイヤ

X1/X1turbo用 5D版2枚組 7,800円
マイクロキャビン ☎0593(51)6482

ーそっちのけで勝手にブライアンは会話してしまうのだ），行動や状況を示すメッセージが表示される。

私は本ゲーム唯一にして最大の欠点がこの豊富で冗長なメッセージであると断言しよう。それについてはのちのちじわじわと述べるつもりである。

さて，以降，直接ゲーム進行に立ち入った話が続出する。その前に，私はカーマインをやるにあたって，心構えの二者択一を迫らねばならない。あなたは“ただひたすらゲームを解くためにプレイする”のか，“戦闘メカに乗ってのさまよい歩きを楽しみたい”と願うのか。前者と後者ではまるっきりゲームに対する接し方が変わってしまう。そんじょそこらの“敵を殺せばいいだけよRPG”や“アタッチ”なアドベンチャーゲームだと，まずは後者，途中から前者と誰でも自然に使い分けられるのだが，カーマインではアドベンチャーゲームとは思えないほどいい雰囲気をもっているので心構えの違いは大きい。

私はなににおいても楽しむことを第一義とするので後者の視点から入る。

いざ，研究所のなかへ

私（≒ブライアン）はコクピットに座り，ガレオスのD.C.（RPGでいうHP。これが0になると死ぬ）をチェックした。

私（≠ブライアン）はひねくれているので，まっすぐ進むのはいやだとばかりに後ろを向き，来たはずの道を戻った。

侵入したエレベーターへ向かう途中，味方であるキースのリュコンに会った。味方に出会うとガレオスが“FRIENDLY”と教えてくれるので楽だ。彼はなにやらおびえたふうである。私（≠ブライアン）は面白がって近寄った。キースは「ここはおまえの担当区域じゃないだろ」と叫んだ。迷わず近寄る。キースの顔が歪む。

「おまえブライアンじゃないな……」

キースは私（≒ブライアン）をバイオドールと思い込んでいる。あっ。撃ってきた。ええい，あんな奴，俺（≠ブライアン）の仲間じゃねえ。といって応戦する。

などとやっていては一生終わらない。怖がっている味方を殺すなど言語道断。勝つには勝ったがD.C.は減るわ，バズーカは壊れるわでろくなことはない。ふりだしへ。

おっ。なにかが近づいてきた。あれ，“>機能停止”。こんなところで殺されるのか。いや，そいつは若い男で，ガレオスに近づいてまた去っていった。バイオドールか？と私（≠ブライアン）は思った。機能が回復したのでスクリーンを見ると，奴は通路を右に曲がった。あとを追ってやろう。

こうして，長い話は進み，敵のAFWと戦ってぼろぼろになりながら工作室の端末を発見。アクセス。データを引き出す。訳がわからないが，一応メモ。終わるとカードBを吐き出したので取る。次の通路へ。

おや，女の子がこちらのメカドールに襲われている。こちらのデータバンクには開発主任の娘“アイリーン・ケストナー”とある。救わなければ。あれ，メカドールはいうことを聞いてくれない。ええい，戦え。そして，敵AFWとの戦いで疲れたガレオスは味方メカドールにやられ，D.C.が0となり，動けなくなった。

“本機はこれより最終行動へ入る”

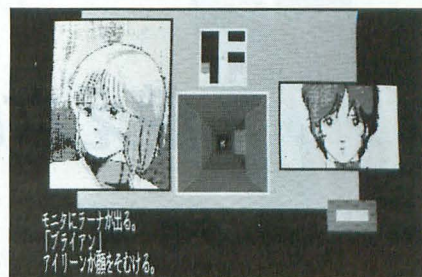
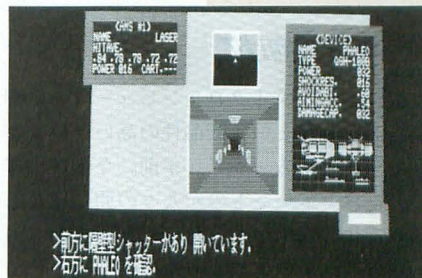
おっと，終わってしまった。私（≠ブライアン）はゲームオーバーといわないところがさすが，と思ったのであるが，終わったものは終わったのだ。やり直しである。

女の子を見捨てて，一度通路へ退き，もう一度出てくる。なんと，女の子ひとりになっているではないか。ラッキー。助ける。ええい，ブライアン（≠私）が勝手に喋っておる。うっとうしい奴だ。だいたいこれが20歳だと。どう考えても15,6の顔ではないか。さてはこの女，バイオドールだな，と私（≠ブライアン）。むちゃくちゃだ。

このようにセーブ・最終行動を繰り返し，やがて小型エレベーターの前に来た。

よし，エレベーターに乗って逃げよう。こんな作戦，付き合ってらんねえや，と私（≠ブライアン）。美人メカドール“LUNA（ラーナ）”が止めるのも聞かずガレオスを降りてエレベーターに乗る……。敵前逃亡だとかいって殺されてしまった。どうも，変なことをするとそれ相応の報いがあるようだ。

やがて，マーコス班のタコとカミカゼに会う。2人ともいかにもアニメに出てきそうなキャラで大笑い。でも，2人とも死んじゃった。合掌。許せないのが，メッセージスクリーン。彼らに会って会話した途端，“君の口調はアイリーンに対してのそれと違って気安い”だと。大きなお世話だ！

終盤，生命維持システム室の前で後ろに私（≒ブライアン）の元恋人“ラーナ”を載せたアレード少佐のガレオスに出会う。おう，勝手にうつむき合うんじゃねえ，アイリーンとラーナ。私（≠ブライアン）はラーナのほうがいいんだい。でもストーリーは変わらない。ブライアン（≠私）は叫んだ。「少佐！ この作戦に命を賭けるほどの価値があるんですか!?」

伝統的三角関係？

シャッターの向こうに敵出現

アドベンチャーはアドベンチャー

ビジュアル効果があまりによくできているせいか，ついついアドベンチャーゲームということを忘れてしまう。シャッターが開くシーン。攻撃時の映像。D.C.。ガレオスとの会話。まるでRPGだ。しかし，レベルアップなんてしない。われわれはアムロでもカミーユでもないのである。このゲームがストーリーにとらわれたアドベンチャーゲームでなかったら，ロボットアニメが文字どおり体験できたろう。特に夜中，闇と静寂に包まれてカーディナルをさまよったらもうそこは別世界。

でも現実にはアドベンチャーゲームで，ストーリーに反したことをするとどこかで死ぬようになっている。イベントが起きるとプレイヤーの手を離れてブライアンが勝手に話を進めてしまう。せめてメッセージをもっと簡潔にするか（“絶句する君”などという文は不要なのだ）省略可能にするか（何度も同じメッセージを読む気にはならない）してほしかった。

選択の幅はどこを歩くか，戦いをするか逃げるかくらいで，あとはあまりない。それでもアクロバット的な謎解きはなく，どんでん返しもよくできていて，なおかつ簡単ではないというきわきわのバランスをもった不思議なゲームである。

私は，こういったいい意味で凝ったゲームの出現を待っている。返す返すも冗長でうっとうしいメッセージと臭くてダサい台詞が残念であった。

THE SOFTOUCH よりよいソフトウェア環境のために〈1〉

使えない人が使うマシン

Tama Yutaka
多摩 豊

ソフトウェアがより進化するためにはなにが必要となるだろう。そんなことを考えつつ，このコーナーではソフトウェアのさまざまな形態を見ていくことにする。まず今回はお膳立て。ちょっと重いがヒューマンインタフェイスから考えてみよう。

次の10年に向けて

混沌としたマイクロコンピュータの世界から，AppleIIという商品性の高いマシンが生まれたのがちょうど10年前。今年1987年はパソコンというものが人々の間に普及し始めてから10年がたったという年である。

最初は汎用コンピュータの小型版として登場したパソコンも，いまでは“知的創造作業”に欠かせない道具となり始めてきている。しかし，この10年の間にパソコンは，本当に誰にでも使える道具に変貌をとげたといえるのであろうか？

現在のパソコンは，グラフィックや音声などのハードに近い部分では日々改良が加えられているけれど，ソフト（といってもプログラムだけを指してるわけじゃない。インタフェイスやデータの扱い方などの“設計思想”のこと）に関しては，まだまだ努力不足，旧態依然としている。

いまやコンピュータの知識がない人間でも，パソコンを操作しなければいけない時代。これからより多くの人に受け入れられていくためには，使う人の種類や使い方によって，作り方を変えたパソコンというものが必要なのではないだろうか？

パソコンを操作する？

コンピュータを動かすシステムのことを“オペレーティングシステム”といい，コンピュータと人間の接点のことを“マン・マシン・インタフェイス”という。

コンピュータを操作するということは，結局“マン・マシン・インタフェイス”を介して，“オペレーティングシステム”を動かすということ（つまり，キーボードからコマンドを打ち込んだりすること）にほかならない。

いまのパソコンは，ここがまずなってない。

AppleII などの当時のパーソナルコンピュータは，コンピュータを操作する命令を組み込んだ“簡単な”言語BASICを搭載して売られた。

しばらくして，パソコンにディスクドライブが当たり前に付いているようになると，オペレーティングシステムも，ROMに組み込まれたBASICではなく，ディスクで供給されるDOS（ディスクオペレーティングシステム）が主流になってくる。代表選手はなんといってもMS-DOSマシン（PC-9800をはじめとする日本のたいていの16ビットパソコンや，IBM-PCなんか）で，こいつはいまだにパソコンの世界を制覇している。

さて，ここまでの流れは，すべてパソコンを大型のコンピュータに近づける方向で進んできている。

プログラミング言語にしろDOSにしろ，大型の機械で使われていたもの（つまり，専門家たちの“おもちゃ”）を，パソコン用に小型軽量化したものであって，まったく知識がなくて，おまけにそんなことを勉強しようって気もない人間には，すごく不思議なシロモノなのである。

たとえば，あるディスクの中に入っているファイルを見てみたいとしよう。これはすごく基本的な操作ということになるけれど，これを行うためにはFILESとかDIRとかの魔法の呪文と文法を知っていなければならない。おまけにそいつを1文字たりとも間違えずにキーボードから打ち込まなければいけないのである。

要するに，ものすごく基本的なことをするためにも，とにかくマニュアル丸1冊（下手すりゃ2冊）と戦わなければならないことになる。

普通の家庭電気製品で，分厚いマニュアルを読んで，両手・両足の指に余る数のコマンドを覚えなければ使えないものがあるだろうか？ 電子レンジにしろ，洗濯機にしろ，“マイコン”が入っている家電製品というのは，その“マイコン”が面倒臭い操作を全部やってくれますってのが売り物なんだから，本家本元のパソコンだって，“操作はお任せワンタッチ”みたいなものもあっていいんじゃないだろうか？ コンピュータを家電製品と一緒にするのが無理だって気もするけれど，それにしたっていちいちカンマやスラッシュの付け方まで細かく指定して，間違えずにキーボードを打たなければ使えないなんてのはひどすぎる。もう少し工夫のしようがあるはずだ。

ところが，実際になんとか改善されている例というのはじつに少ないのである。

遅いマシンはなぜ評価されないか？

“コンピュータは，より速く，より高機能でなければならない”

これは確かに真理かもしれない。誰だってスピードの遅いパソコンよりは速いほうがいいに決まっている。

ところが，ここで困っちゃうのは，ちょっと使いづらくても，より速い機械を作っていれば受けるはずだって，ハード屋さんが考えているということである。あるいは，機能が高ければ多少の使いづらさは気にならない（気がつかない？）人たちが使いやすさというものを考えているといったほうがよいかもしれない。

スピードを追求するというのは，たとえばレーシングカーを作るのと同じことである。ところが，F1マシンを普通のドライバーが運転できないのと同じことで（なにしろ，ギアが堅くてギアチェンジすらムズカシイらしい），パソコンも普通の人にはそれ用に，親切で簡単な設計をしなければ使ってもらえないのである。

コマンドをいちいちキーボードから打ち込むよりは，使えるコマンドをメニュー形式で表示して，それを選ぶようにしてやったほうが使いやすいに決まっている。なにしろコマンドを暗記しなくてすむ。

ファイルの名前の付け方にしても，ファイル管理のための情報はコンピュータが勝手に付けてくれて，ユーザーは自然の言葉に近いような名前の付け方ができるほうがいい。ちょいとオペレーティングシステムを工夫すれば，その文書がたとえば“一太郎”の文書ファイルなのか，バイナリファイルなのかなんてのは使ってる側にわからないような形で管理できるはずだ。そうすれば，ファイルに名前を付けるときや，ファイルを探すときに，面倒な“名前の付け方”の規則を知らなくてもすむ。

たとえば“GENKOU/MZ/9.DOC”なんて名前より，“Oh！MZ８月号原稿”のほうがわかりやすい。そのファイルがどんなソフトの文書であるかなどということは，たとえばアイコン（絵文字）で表示すればいい。ソフト自身のアイコンと，そのソフト用のファイルのアイコンが似ていれば，誰にでもすぐわかる（もちろん，こういったことを実現するためには従来のシステムとの互換性を切り捨てなければならないが）。

余分な（というか専門的な）情報は，極力コンピュータに管理させて（つまりそれだけシステムは大きくなる）使う人は作業に専念する，これが使いやすいということの基本だと思う。

ユーザーフレンドリーな（要するに，素人にも使えるかもしれない）オペレーティングシステムは，一般的に複雑で大きなものになる。そうなるとマシン自体の速度は遅くなる。おまけに，せっかくのオペレーティングシステムを使いやすくしておくためには，むやみやたらに機能も増やせなくなる（ユーザーの使いそうもない機能まですべてサポートするわけにいかなくなるからね）。

これまでのパソコンユーザーにとっては“かったるくて，遅い”マシンになるかもしれないけれど，そういう人は，それこそ“F1マシン・パソコン”でも使っていればいい。そういうパソコンなら心配しなくてもいくらでも出てくる。

ハードのスピードが上がって，中身が複雑でユーザーにとっては使いやすいパソコンを作れる時代になったのだから，いままでとまったく違う人種を対象としたこれからのパソコンは，操作方法も１から作り上げていかなきゃいけないはずである。

それではなにが必要か？

では，いったいどういうパソコンが本当に必要なのであろうか？（もちろん，これをズバッといい当てられるぐらいなら，もうとっくに誰かがやっているだろうけど）。

AppleのMacintoshや，日本のTRONプロジェクトなどで進めているマン・マシン・インタフェイスの方向性は，“What you see is what you get”という考え方に基づいている。

この“WYSIWYG”とは，早い話が，自分がコンピュータ上でやりたい仕事を，あたかも机の上でやっているのと同じように目に見える形にするという考え方で，たとえばファイルを削除したければ，そのアイコンをゴミ箱アイコンに持っていって捨てるとか，文書の中で切り取りたい部分は，マウスで指してやればよいとか，できうるかぎりコマンド入力を排除した操作方法なのである。

X68000のヒューマンインタフェイスも，この路線を狙っているようだが，どうやらMS-DOSとの関係を重視しているおかげで，この考え方を徹底できなかったようだ。

たとえば話題のビジュアルシェルだが，ワープロで文書を登録すれば，ワープロ文書であることを示すアイコンが自動的に付く（拡張子.SWPも自動的に付いて管理される)。このアイコンをダブルクリックすれば，「オレはワープロの文書だ」と自己主張してワープロ（WP.X）を立ち上げるのだ。ただし，アイコンの下にはやっぱりMS-DOS流のファイルネームが表示されている。MS-DOSとのファイルの互換性をもつためにはしかたのないことらしい。いっそ拡張子なんて見えなくしちゃったらどうだろうかとも思うのだが，そうするとワイルドカードが使えないじゃないかといった声も聞こえてくる。結局，X68000のビジュアルシェルが狙っているのは，コマンドモードで使いこなせる人にとっての「よりいっそうの便利さ」であるらしい。

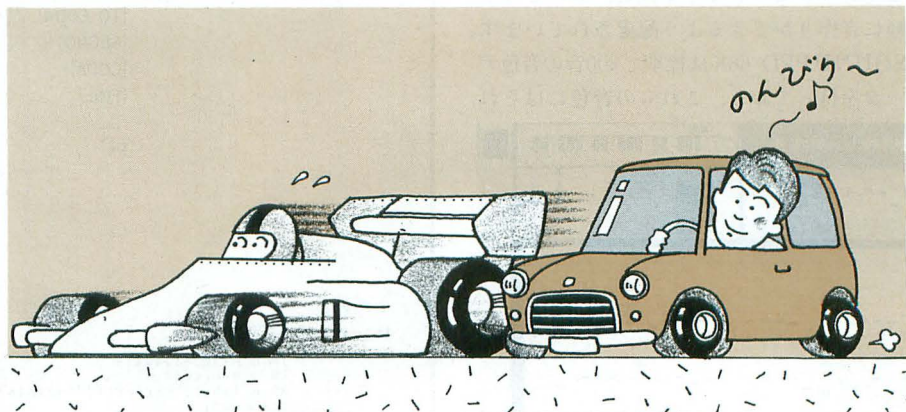
ビジュアルシェルとコマンドモードのどちらが立ち上がるかをユーザーが選択できるのは評価できる。だけど，もしなにも知らない人が“TITLE.SYS”をほかの名前に変えちゃったら……結局マニュアルをひっくり返す羽目になる。もし誰にでも使えるシステムを目指すならもっと別の配慮も必要ではないかと思う。

“スピードを誇る”マシンではなくて，“スピードに裏打ちされた使いやすさ”を目指すなら，小学生ぐらいの子供が遊んでいるうちに使い方を覚える，それぐらい簡単でなければいけない。

MS-DOSの使い方にこだわる必要はない（とはいえ，MS-DOSのファイルは読めたほうがいい)。MS-DOSへ行きたい人はどうせパソコンに慣れた人なんだから，そういう人のことまで心配する必要はないのである。

その機械の主なユーザーが研究にコンピュータを利用する人なのか，仕事でソフトウェアを使いたい人なのか，それともコンピュータを趣味に利用する人なのかによってオペレーションシステムも違っていいはずである，いや違わなければいけない。

パソコンを作るときにそこまで割り切って作ってほしい。そうすれば，自然と特徴があって，使いやすいマシンというものも生まれてくるはずである。

要するに，ヒューマンなインタフェイスは，まず“誰のためにどんなものを作るか”を見極めることから始まるのである。

X68000

SOUND PRO 68K/SWITCH.X/スプライト活用法

X68000あなたの知らない世界

編集室

まだまだX68000には謎がいっぱい。そして夢もいっぱいです。今回は皆さん待望のスプライト関数を公開しましょう。いまならみんな初心者マーク。X68000に関する疑問点などもこのコーナーにお寄せください。

ようやく市販ソフト第1弾のゼビウスが発売されひと安心といったX68000のソフトウェア状況ですが，現在のところPC-9801のグラフィックツール，FM77AVのミュージックツール，Macintoshのビジネスソフト，アーケード版のゲームと各界の一流どころがグレードアップして結集しているようです。

SOUND PRO 68K(CZ-214MS)

すでにシャープの広告にも姿を見せていますが，これはいうまでもなく8重和音のFM 音源を自由自在に操るためのツールです。X-BASICのMMLはそれ自身で制御構造を持つなどなかなか賢いものですが，バンドリングされている音色は68音とあまり多くありません。このツールはありとあらゆる方法で自分だけの音を作る手助けをしてくれます。このソフトはFM77AVシリーズのミュージックソフトを手がけたミュージカルプランが担当しています。(発売はシャープ)。このソフトの**正しい**価格は15,800円で7月末発売予定となっています。

さて，SOUND PRO 68Kですがフルマウスオペレーションは当然として，VIPのようにOPMのレジスタに直接値を指定するモード，音色のイメージによって「明るい音で立ち上がりは遅く，ビブラートは少なめに……」というぐあいに音色を設定するイメージモードなどを備え，誰にでも手軽に音作りができるよう配慮されています。SOUND PRO 68Kは標準で200音の音色データを持っており，これらの音色にはそれぞれの音色を生かしたサンプルミュージックデータが付属します。これらのデータは新しく音色を作るときにも使用されます。つまりミュージックデータを演奏しながら，ある音色のAR(アタックレイト)だけを連続的に変化させてみたりといった操作も自由自在です。

X68000がシンセに変身

さらに，こうしてエディットされた音色は波形を3D表示することで視覚的に確認することもできます。この機能はフェアライト社のプロ用シンセサイザに搭載されて有名になったものですが，SOUND PRO 68Kでは本家よりも高速の表示を達成しています。当然これらのデータはBASICの音色ファイルとして使用することが可能ですし，追って発売が予定されているミュージックエディタMUSIC PRO 68K (CZ-213MS：18,800円）でも使えます。MUSIC PRO 68Kでは楽譜をそっくりそのまま入力できる楽譜ワープロ機能，コードネームとリズムパターンの指定による自動伴奏機能などを備えたこれまでにない画期的なツールとして登場しそうです。グラフィックツールのZ'sSTAFF PRO 68Kとあわせ，まさにプロ用のツールが7月から8月にかけてどっと勢揃いの予定です。

"音を見る" 3D表示モード

史上最強のMML

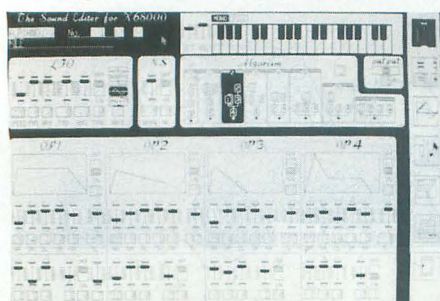
繰り返し処理などをサポートし，かなり高機能なX68000のMMLですが，それでも全8パートのデータをミュージックトラック上に置いて演奏をするのですからメモリの使用量はばかになりません。楽譜にちょっと入り組んだ繰り返し部分があるとそれを展開してトラックに入れなければなりませんし，またステレオで音を鳴らそうとすると3つ分の音色を設定しなければなりませんでした。

しかし，どうやらもっと便利なものが隠されているようです。表1がマニュアルに記載されていないMMLのコマンドです。なんとD.C.やD.S., Coda, Fineなど，お馴染みの音楽記号がそのまま使用できるのです。こういった利口な命令を駆使することでデータ量を大幅に減少させることができそうですね。

表1
MML(MUSIC MACRO LANGUAGE)追加分

文字	略	意味
[D.C.]		先頭に戻る
[D.S.]		[SEGNO]にとぶ
[TO CODA]	[*]	[CODA]にとぶ
[SEGNO]	[$]	[D.S.]からのとび先
[CODA]		[TO CODA]からのとび先
[FINE]	[^]	この記号のある小節で[SEGNO]など繰り返した演奏が終わることを示す
Px		xの値によってLEFT, RIGHTに音を振り分けたり，L•R同時に出力したりします 1……LEFT ONLY ON 2……RIGHT ONLY ON 3……L•R ON

リスト1
サンプルプログラム

```
10 m_alloc(1,1024)
20 m_assign(1,1)
30 m_trk(1,"cegceg[*]dfadfa[$]cdefg|:ceg|1dfa:||2:||3ceggag[d.s.]dfadfa[d.c.]
[coda]cccc")
40 m_play(1)
```

SWITCHの使い方

皆さんはHuman68kでswitchコマンドを使ったことがありますか？　ビジュアルシェルでswitch.xをダブルクリックするとわけのわからない文字列が出てきて，あわててブレイクキーを押した人もいるのではないでしょうか。Human68kのユーザーズマニュアルによればswitchコマンドは以下のような機能を持っているとされています。

RS-232Cの設定
メモリサイズの設定
起動デバイスの指定
電源断時のディスクイジェクトの有無
OPT.2キーの設定
画面のコントラスト
キーボードのカナ配列の方法

しかし，このコマンドはそれだけではありません。ここは黙ってコマンドモードから，

switch ?

と打ち込んでみてください。switch.xのコマンドメニューが表示されましたね。最初の7つ目まではマニュアルに書いてある用法ですが，そこから下の命令はマニュアルに記載されていませんのでそれらについて説明します。

―Tv_ctrl

このコマンドは本体の電源を落としたときの動作を指定するものですが，コマンドメニューを見る限り電源のoffテレビモードしか選択できないようにも思えますが，そこはテレビ事業部，あらゆるコントロールを可能にしています。ちょっと7月号を見てください。X68000のIOCSコールが載っていますね。IOCSコールの$0CにTVCTRLというのがあります。このコマンドではその入力条件のD1レジスタにパラメータを入れます。

つまり，$03を指定すると電源は落とさずにテレビのボリュームをふつうの状態にして終了するのです。もとに戻したいときは$0Dを指定します。

―Lcd_mode

Lcdとは液晶のことですが，別にX68000に液晶ディスプレイがつくわけではありません。OSやBASICなどではOPT.1＋OPT.2キーで電卓を使うことができますね。このコマンドはそのときの文字フォント選択を行うものです。Lcdを指定すると7セグメントの液晶っぽい文字に，Normalを指定するとふつうの漢字ROMに登録されている文字になります。

―Sram

Sramは容量こそ小さいものの，メモリがバッテリでバックアップされているという利点があります。やろうと思えばここにプログラムを置いて起動時に自動実行させることも可能です。そしてこの部分を使用するためのコマンドがSramです。

しかしRAMディスクとして使用するには，これだけではだめでCONFIG.SYSにDEVICE＝SRAMDISK.SYSという1行を書き加えねばなりません。

―P0～―P8

これらは，テキストパレットのデータを指定するものです。もちろん，65536色モードのカラーコードを用います。P0からP3はBASICのCOLOR［　］文で指定するP0からP3と同じです。P4とP8はVSで使っている色のような気もするんですが，ちょっと違います。開発中に変更されたのかなあ……結局，よくわかりませんでした。

―Db

このコマンドはROMデバッガを自動起動させるものです。ただし，これを使用するためにはRS-232Cを持ったコンピュータとRS-232Cのクロスケーブルが必要です。具体的な使用方法としては，まず2台のコンピュータを接続し，一方をターミナルモードで起動したあとROMデバッガをONにしたX68000を起動します。するとターミナルの画面上にROMデバッガの起動メッセージが表示されるのです。あとはバグが発生するまで待つか，インタラプトスイッチを押したりするとターミナル側に制御が移ります。Hキーでヘルプ画面が表示されますのでプログラム開発に役立てることができるでしょう。

―Xchg

このコマンドはキャラクタセットの一部を変えるものです。具体的にこのコマンドで変更されるキャラクタは￥，￣，｜の3つです。これらはXchgで指定する値のビット0から2が1ならそれぞれ＼，～，¦に変えてしまうものです。当該ビットが0のときはもとのフォントに戻ります。これでC言語のソースプログラムを美しく記述できますね。

―Hd_max

あまり説明する必要もないと思いますが，ハードディスクの最大ドライブ数を設定するものです。

―Wait_prn

これは本体がプリンタになにかを出力する場合に，プリンタに信号を送ったあとプリンタから信号を送ってくるまで待つ最大の時間を設定するコマンドです。間違えてCOPYキーを押したときなどの待ち時間がうっとうしければ短めに，カットシートフィーダなどをお使いの場合は長めに設定しておくとよいでしょう。

―First_ky, Next_key

この2つのコマンドはキーを押してからキーリピートが始まるまでの時間とリピートの速度を設定するコマンドで，コマンドメニューにその時間を求める式が書いてあります。

表2　SWITCH.Xのコマンドメニュー

```
***** SWITCH Version 1.00 使用法一覧 *****
-Rs232c  =9600bps Bits-8 Parity-None Stop-1 Xon
-Memory  =1024k         2048k/・・・
-Boot    =STD           HD0~HD15/2HD0~2HD3/ROM$?/RAM$?
-Eject   =OFF           ON
-Opt2key =Tvctrl        Normal
-Contrast=14            0~15
-Kana    =Jis           Aiu
-Tv_ctrl =Off           Tv
-Lcd_mode=Lcd           Normal
-Sram    =No_use        Ramdisk/Program
-P0      =$0000         >
-P1      =$f83e         >
-P2      =$ffc0         >テキストパレットデータ
-P3      =$fffe         >
-P4      =$de6c         >
-P8      =$4022         >
-Db      =OFF           ON(ROM_DBの自動起動)
-Xchg    =0             bit0-2 = ￥￣|
-Hd_max  =1             HDのドライブ数(0~15)
-Wait_prn=$80000        プリンタータイムアウト
-First_ky=3             0~15(100*n+200ms)リピートまで
-Next_key=2             0~15(5*n^2+30ms)リピート間隔
*** 値は現在の値ではなく、標準の値です ***
-
```

表3　ROMデバッガのヘルプメニュー

```
   00FF1B0A       move.w  #$FFFF,$00000A0E
   00FF1B12       bra.w   $00FF057A
   00FF1B16       addq.l  #8,A7
+H
A                 :assemble              R address,drive,sector,length
B                 :display breakpoint                        :read device
B[bp][address]    :set breakpoint        S[size]<range> data :search memory
BC[<bp>]          :clear breakpoint      T[=address][count]  :trace
BE[<bp>]          :enable breakpoint     U[=address][count]  :untrace
BD[<bp>]          :disable breakpoint    W address,drive,sector,length
BR                :reset break count                         :write device
D[size][<range>]         :dump memory    X         :display register
E[size][address]         :edit memory    X[reg]    :register change
F[size] <range> data     :fill memory    Y/N       :yes no ask
G[=address][adress]      :go             Z[num=exp] :system variable
H                        :display this   ?[exp]    :print expression(hex)
HI                       :trace history  ??[exp]   :print expression(dec)
L[<range>]               :list           ￥        :loop command line
M <rang> address         :move memory

operators  + - * / & (and) | (or) ! (not) % (residue) ^ (exor)
number     ??(hex.) ￥??(dec.) _??(bin.)
size       s(byte) w(word) l(long)
+
```

以上のようにswitchひとつでもなかなか奥が深いものですが，これらのコマンドのなかにはあまり不用意に触らないほうがよいものもあります。特にBoot関係には手を出さないように，最悪の場合ユーザーには修復不可能な状態になることもあります。気をつけてください。　(加藤賢哉)

スプライト活用法

さてX68000の強力な機能のひとつにスプライトがありますね。BASICマニュアルにはごく簡単にしか解説されていませんがX68000のスプライトはおそるべきポテンシャルを秘めています。そもそも，このスプライト表示を行っているCYNTHIAというカスタムLSIにはX1turbo3台分に相当する回路が詰め込まれているのですから，グラディウスだろうがゼビウスだろうが余裕でこなしてしまいます。今回はBASICでスプライトを扱ってみましょう。

理論編

基本的にX68000のスプライトは128枚のスプライトプレーンと2枚のバックグラウンドプレーンを持っています。スプライトプレーンはグラディウスの空中キャラクターなどに使用されているもので表示位置を指定するだけで簡単，高速にキャラクターを表示してくれます。このプレーン1枚にはひとつのスプライトが書き込まれますので同時に128個のスプライトを動かすことができるのです。

マニュアルで説明されているのはスプライトプレーンの標準的な部分のみですが，実際にはもっと多くの機能がBASICから使用可能です。表4はスプライト関係のマニュアルに記載されていない関数群です。拡張パターンコードを見てもわかるようにスプライトの表示は単にパターンをそのまま表示するだけでなく，同時にパターンの回転なども指定できるようになっています。ひとつのスプライトパターンは16色で構成されますが，この16色は65536色中16色を選択したパレットから指定されます。X68000はこういったパレットを15個持っていますから画面全体で240色の色を使用できるのです。

これらのスプライトは1024×1024の仮想画面上で座標指定されますが，座標系は画面左上隅が(−16，−16)となるように設定されています。MSXやファミコンのスプライトでは指定座標と表示座標が1ドットずれるといったことがありましたがX68000ではそんなことはないようです。また，ハードウェアの制約により横方向に同時に設定できるパターンは32個に限られています(仮想画面上)。しかし，32個のスプライトといえば表示画面上で1列に並べると512ドット，高解像時の画面横ドット数に相当します。これだけあれば，そう困ることはないでしょう。

それに対しバックグラウンドプレーン(以下BGと略す)というのはX1のPCG機能とほぼ同じもので，1枚のプレーンに64×64個のパターンを表示します。このBGは512×512モードでは1パターンが16×16，256×256モードでは1パターンが8×8ドット構成となります。使用できるBGプレーンの枚数は256×256モードで2枚，512×512モードで1枚，ただしスプライトパターンを128個以上定義する場合，BGプレーン0は正常に表示されません。

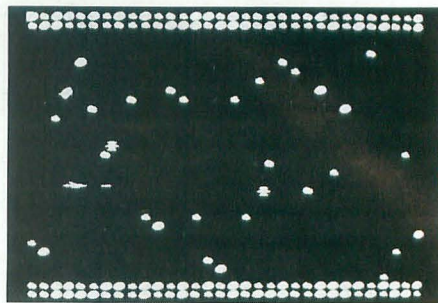
BGを扱う際は直接BGプレーンを指定する場合と各プレーンを割りつけたテキストページにより指定する場合があります。

ほとんどPCGと同じように使えるのですが，PCGと違うのはこれらのプレーンが2枚独立にスクロールできること，ひとつのパターンにつき16/65536色が指定できることなどです。BGは高速でキャラクターを動かすことにはむいていませんが（できないこともない）横方向32パターンという制限もないため，その名のとおり背景として用いるのがよいでしょう。

実技編

では，実際にサンプルプログラムを見てみましょう。このプログラムはキャラクター定義などを簡略化するためグラディウスのスプライトデータをそのまま使用しています。まず，グラディウスを立ち上げゲームが始まるとすぐにブレイクキーを押してください。このまま電源を落とさずにBASICを立ち上げるとグラディウスのデータをそのまま使うことができるのです。これでデータ部のプログラムを削除することができました。

画面上の岩石群はBGプレーンの0番に描かれています。200行から390行までが背景を設定している部分です。このプログラムではあらかじめ仮想画面いっぱいに背景を描いておき，hmの分量だけスクロールさせています。

430行以降がメインルーチンです。このプ

BGを使ったゲーム画面

ログラムではマウスを使用して自機を操作しています。MSPOSは直接マウスカーソルのX，Y座標を返すので非常に簡単な書式で自機を移動させることができるのです。自機の表示ルーチンは通常の場合，470行から510行，破壊された場合は520行から560行となっています。

570行から690行が敵の移動ルーチン。変数lim+1が画面上に現れる敵の総数です。当然，敵の数を増やすとそれだけ遅くなってしまいます。700行から760行がミサイルの処理，770行から790行が当たり判定を行っている部分です。

「BASICでゼビウスが書けるか？」という期待もあったのですが，BASICの処理速度は思ったほど速くありません。制御構造をGOTO文に展開しマルチステートメントを多用するというおきまりの手法を使えば速くなるのですが，X-BASICという言語はCへのコンバートを前提にされた言語ですしX68000自体がきたるべき32ビット時代を指向したマシンなのですからへたな小細工は無用です。コンパイルすればいくらでも速くなるのだからCへコンバートできないプログラムは書くべきではないでしょう。

おまけ

先月バッチファイルの使用法をお話ししましたが，VS上からバッチ処理を行うにはアイコンメンテナンスでファイル名の部分をクリックし，

　*.BAT

というファイル名を入力してアイコンを呼び出し，そのままOKで登録してください。詳しいことはわかりませんが，以上のようにアイコンメンテナンスで再定義をすればデ次回からはダブルクリックでバッチ処理ができるようになります。　(中野修一)

周辺機器速報

X68000用ユニバーサルI/Oボード（CZ-6BU1:39,800円）が発売されました。また2Mバイト，4Mバイトの拡張RAM(CZ-6BE2:79,800円，CZ-6BE4:138,000円）も7月末発売予定となっています。

リスト2　サンプルゲーム

```
 10 screen 1,3,1,1:console ,,0
 20 locate 0,10
 30 int lim=3,gx=55,gy=247,gb=0
 40 int mx,my,ml,mr,x(99),y(99),t,r,e,w,c
 50 char cl(255),ce(255):sp_pat(0,ce)
 60 r=0:for t=0 to 15:r=r+ce(t*t):next
 70 if r<>0 then {
 80      sp_def(0,cl)
 90      for t=32 to 95:sp_pat(t+96,cl)
100      sp_def(t,cl):next:beep}
110 bg_set(0,0,0):bg_fill(0,0)
120 for t=0 to lim
130      x(t)=rand() mod 496
140      y(t)=16+(rand() mod 464)
150 next
160 for t=0 to lim+2
170      sp_move(t,0,0,0)
180 next
190 mouse(2):mouse(4)
200 for t=0 to 29
210      mx=rand() mod 32
220      my=(rand() mod 28)+2
230      e=(rand() mod 3)+376
240      bg_put(0,mx,my,e)
250      bg_put(0,mx+32,my,e)
260 next
270 my=0
280 for r=0 to 1
290      for t=0 to 31
300           e=376+(rand()mod 3)
310           bg_put(0,t,0+r,e)
320           bg_put(0,t+32,0+r,e)
330      next
340      for t=0 to 31
350           e=376+(rand()mod 3)
360           bg_put(0,t,31-r,e)
370           bg_put(0,t+32,31-r,e)
380      next
390 next
400 sp_off():sp_on(0,lim+3)
410 sp_disp(1):bg_set(0,0,1)
420 /*80
430 while 1
440      c=c+1:msstat(w,w,ml,mr)
450      hm=(c mod 128)*4:bg_scroll(0,hm,0)
460      w=1
470      if gb=0 then {
480           mspos(gx,gy)
490           sp_move(0,gx,gy,65)
500           sp_move(1,gx+16,gy,66)
510           w=0}
520      while w:w=0
530           sp_move(0,gx,gy,95-gb)
540           gb=gb-1
550           if gb=0 then end
560      endwhile
570      for t=0 to lim
580           sp_move(3+t,x(t),y(t),32+(c mod 5))
590           e=gx-x(t):w=gy-y(t)
600           if abs(e)<16 and abs(w)<16 then mr=1
610           x(t)=x(t)-16
620           if x(t)<-15 then {
630                x(t)=512
640                y(t)=16+(rand() mod 464)}
650           e=mx-x(t):w=my-y(t)
660           if abs(e)<16 and abs(w)<16 then {
670                sp_move(3+t,x(t),y(t),72)
680                x(t)=-16:mx=510 }
690      next
700      w=1:if my=0 then {
710           if ml then my=gy:mx=gx+16 else{
720                     w=0}}
730      while w:w=0
740           mx=mx+16:if mx>511 then my=0
750           sp_move(2,mx,my,77)
760      endwhile
770      if bg_get(0,j(gx)/16,(gy+8)/16) then mr=1
780      if bg_get(0,j(gx-16)/16,(gy+8)/16) then mr=1
790      if mr then if gb=0 then gb=11:sp_off(1)
800 endwhile
810 end
820 func j(a)
830      return((a+32+hm) mod 1024)
840 endfunc
```

表4　スプライト関数追加分

sp_init (　)

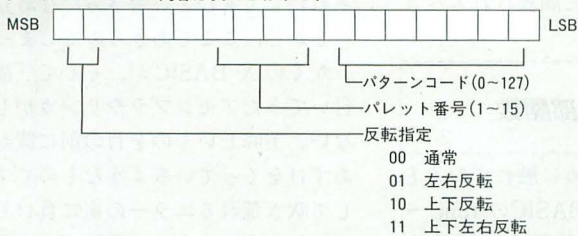
引数　なし
処理　スプライト面の初期化をします
戻り値　int　常時　0
備考　スクロールレジスタ，PCGパターン，スプライト用パレットなどの初期化

sp_set (char1, [int1], [int2], [int3] [, char2])

引数
- char1　スプライト番号 (0〜127)
- int1　Xポジション (0〜1023)
- int2　Yポジション (0〜1023)
- int3　拡張パターンコード (V, H, COLOR, CODE) (0〜&hfcff)
- char2　プライオリティ (0〜2)

処理　スプライトレジスタ（ポジション，パターンコード，パレットブロック，プライオリティなど）の設定をします
戻り値　int　常時　0
備考　XポジションおよびYポジションは，画面左上端を座標（16，16）とする拡張パターンコードの内容は次のとおりです

MSB ［16ビットのボックス］ LSB
- 反転指定
- パレット番号 (1〜15)
- パターンコード (0〜127)

反転指定
- 00　通常
- 01　左右反転
- 10　上下反転
- 11　上下左右反転

プライオリティは次のように指定します
- Pri=0　スプライトを表示しない
- =1　BGより後ろに表示
- =2　BG0より後ろ，BG1より前に表示
- =3　BGより前に表示

sp_stat (char1, char2)

引数
- char1　スプライト番号 (0〜127)
- char2　読み出しデータの選択 (0〜3)
  - =0：Xポジション
  - =1：Yポジション
  - =2：拡張パターンコード (V, H, COLOR, CODE)
  - =3：プライオリティ

処理　スプライトレジスタ（ポジション，パターンコード，パレットブロック，プライオリティなど）の読み出しをします
戻り値　int　読み出しデータ

bg_scroll (char, [int1] [, int2])

引数
- char　バックグラウンド番号 (0〜1)
- int1　X座標 (0〜511 or 1023)
- int2　Y座標 (0〜511 or 1023)

処理　バックグラウンド（0 or 1）のスクロール座標を設定する
戻り値　int　常時　0
備考　スクロール座標の範囲は，0から表示画面の大きさの2倍まで
512×512モードでは，バックグラウンド番号1は表示されない

bg_set (char1, [char2] [, char3])

引数
- char1　バックグラウンド番号 (0〜1)
- char2　テキストページ番号 (0〜1)
- char3　表示ON(1)/OFF(0)

処理　バックグラウンドのテキストページ番号割り当て，および表示ON/OFFの設定をします
戻り値　int　常時　0
備考　512×512モードでは，バックグラウンド番号1は表示されない

bg_stat (char1, char2)

引数
- char1　バックグラウンド番号 (0〜1)
- char2　読み出しデータの選択 (0〜3)
  - =0：X座標
  - =1：Y座標
  - =2：テキストページ番号
  - =3：表示ON/OFF

処理　バックグラウンドレジスタ（座標，テキストページ番号，表示ON/OFF）の読み出しをします
戻り値　int　読み出しデータ

bg_fill (char, int)

引数
- char　テキストページ番号 (0〜1)
- int　拡張パターンコード (V, H, COLOR, CODE) (0〜&hfcff)

処理　BGテキストの指定されたページ領域を，第2引数で指定されたデータで満たします
戻り値　int　常時　0

bg_put (char1, char2, char3, int)

引数
- char1　テキストページ番号 (0〜1)
- char2　テキストX座標 (0〜63)
- char3　テキストY座標 (0〜63)
- int　拡張パターンコード (V, H, COLOR, CODE) (0〜&hfcff)

処理　BGテキストの設定をします
戻り値　int　常時　0

bg_get (char1, char2, char3)

引数
- char1　テキストページ番号 (0〜1)
- char2　テキストX座標 (0〜63)
- char3　テキストY座標 (0〜63)

処理　BGテキストの読み出しをします
戻り値　int　読み出しデータ

X68000

X-BASICの外部関数を作る

X-BASICにはマシン語サブルーチンなんて必要ありません。外部関数として組み込んでしまえばいいのです。自分だけのBASICを育てていく。X68000にはこんな楽しみもあるんですね。

Kuwano Masahiko 桒野 雅彦

僕だけのBASIC

X-BASICは，ごく基本的な命令や関数以外は起動時に外部のファイルから取り込むようになっているため，ユーザーの好みに応じてBASICそのものの再構築や関数の追加が行えるという大きな特長を持っています。

従来は，機械語で作ったサブルーチンはBASICやBASICプログラムとは異質なものとしてとらえられており，CLEAR命令でBASICの使う領域を制限してからUSR関数などで直接アドレスを指定して呼び出してみたり，あるいはBASICの空きエリアに機械語サブルーチンを格納し，BASICの予約語テーブルを書き換えてしまうことで，ふだんあまり使わない命令をユーザーの作成したプログラムの実行に使うなど，いずれにしてもかなり不自然な形でしか利用できませんでした。

これに対して，X-BASICは起動時に読み込まれた関数（外部関数）も最初からあるものと同じようにインタプリタの中に組み入れてしまいます。この機能のおかげで，将来新しいデバイスや周辺カードが出たときにも，それらを扱うための外部関数を用意して，BASIC.CNFの中にファイル名を登録しておくだけで，それまでのBASICとなんら使い勝手を変えることなく，新しい機能を十分に活用することができるのです。

自分にとって必要のない関数群は初めから切り離すことも自由です。X68000の場合には標準で1Mバイトのメインメモリを積んでいますが，ハードウェアのパフォーマンスからすると，これでもかなり妥協しながら使うことになりがちです。BASICのフリーエリアも530Kバイトくらいまで確保することができるのですが，「68000なんだから64Kバイトなんてけちくさいことはいわずに……」とかいいながら，多次元の配列を取った場合などではメモリの不足を感じることがままあります。このような場合に，たとえばsinなどの数値演算関係の関数を切り離すことができれば，ちょっとしたワークエリアとしては十分な空間が確保できることになります。

このように自分で好きな環境を構築，設定できるのがX-BASICの大きな特長なのですが，残念ながら例によってマニュアルには外部関数の作り方についてはまったく触れられていません。外部関数の作成はなにもメーカーやソフトハウスなど商売でパソコンしている人だけの特権ではありません。ユーザーレベルでも自分で好きな名前の関数を作ってそれをBASIC.CNFの中で関数として登録してやれば，自分専用の命令を持ったBASICが完成するのです。パラメータの引き渡しも自由ですし，当然のことながら外部関数は機械語ですから，関数が呼び出されたあとはCPUの能力をフルに生かすことができることになります（もちろんそのためには68000の機械語やアセンブラの知識が必要なのはいうまでもありません)。これこそユーザーに開放されるべきものだと思いませんか？

内部関数と外部関数

ここでちょっと念のために触れておくと，外部関数の呼び出し方はBASICのfunc～endfuncで書いた関数（内部関数）となんら変わりません（厳密にいえば上位コンパチ)。名前を書いて，カッコの中で引数を指定します。ですから，BASICで書いていた関数のうちどうしても速度の点で不満なものやアセンブラで書いたほうがすっきりするといったような場合には，その関数だけを外部関数に作り替えれば，ほかはなにもいじる必要はありません。つまり，関数を使う側からすれば，それが内部（いわゆる“サブルーチン”に相当）であろうが外部（“機械語ルーチン”に相当）であろうがまったく考える必要はないわけです。

実行速度はよほどタコな書き方をしないかぎり外部関数のほうが高速化できるでしょうから，ややこしい引き数の計算はBASICで行って，いざ値が確定したら外部関数で一気に処理するという方法を使えば手軽に高速の“BASICプログラムによる”処理をすることも可能でしょう。これまでのBASICでは機械語サブルーチンを利用するにしても常に絶対アドレスを意識しながら無理やりという感じで使っていたのが，X-BASICではそれぞれを関数として名前を付けることでずっとエレガントに，より自然に利用することができるようになったのです。

このようにX-BASICならではの強力な武器である外部関数なのですが，先にも触れたように標準で添付されるマニュアルでは関数の作り方についてはまったく触れられていませんし，またシャープから公開された資料もややわかりにくいものでした。私もその資料を見ながら頭を抱えてしまったひとりです。「まあ，技術屋さんが書いた説明書なんてこんなものかもしれないけど，それにしてもひどい書き方だなあ」。

しかし，ここであきらめてしまってはせっかくのX-BASICが，そして「福袋」に付いてきたアセンブラやリンカがもったいない。美味しいものを目の前に置かれておあずけをくっているようなものです。かくして吹き荒れるエラーの嵐に負けじとつつき回した末にようやく理解できました。わかってしまえばさして難しいことをいっているわけではなく，またそれなりによく考えられた構造となっているのです。

XファイルとFNCファイル

発売前にX-BASICの紹介をしたとき「.Xの拡張子が付いている」と書いたのですが，製品バージョンでは「.FNC」になりました。じつは，XファイルもFNCファイルもアセンブルやリンクのやり方はまったく同じなのですが，Xファイルがそれ自体で実行可能なのに対して，FNCファイルはあくまで

BASICから呼び出されて動くなど，多少性格が異なるのでこうなったのでしょう。

ふつう，アセンブラ/リンカを使って作成したプログラムはそれ自体で完結していますから，ただひとつの実行アドレスがあればよく，このためリンカを通して作成したファイルには実行アドレスが書いてあって(特に指定しなければtextセクションの先頭)，プログラムはそこからスタートするようになっています。ところがBASICの外部関数の場合，ひとつのファイルにひとつの関数しか書けないというのでは，関数の数だけファイルが必要になってしまうなどいろいろとやっかいです。

そこで，ひとつのファイルの中にいくつもの関数を定義しておくことができるような細工が考えられました。ファイルの中の各関数に関する情報を並べたテーブル（ヘッダ）を作り，これをプログラムの先頭に置いておくという方法です。BASICはヘッダを参照して関数の実行アドレスや必要とするパラメータなどを知り，それによって各関数を呼び出すのです。このヘッダを作成しなければならないことと，textセクション以外は使えないことを除けば，ほかは通常の実行ファイルの作成とそれほど変わるところはありません。逆にいえば，外部関数作成の鍵はヘッダにあるのです。

ヘッダ部の構造

それでは外部関数のヘッダ部の構造を見ていきましょう。図1にテーブルの中身と関係を書いてみました。ちょっと難しいかもしれませんが，図を見ながら順に追いかけてみましょう。

先頭から順に11個のアドレスが書いてあります。最初から6番目まではあえていうまでもないでしょう。X-BASICは起動時，RUN，END，SYSTEMなどの命令実行時，そしてブレイクされたとき，特別なことがあったときになんらかの処理をしておく必要があるデバイスのことを考え，それぞれの条件に応じて実行するサブルーチンの定義ができるようにしているのです。たとえば，今回作ったサンプルプログラムではこのような操作は必要ないので，これらが指している先はいきなりrtsになっていて，なにもせずにリターンするようになっています。

7，8番目は将来に備えた予備用の領域です。現在は使用されていませんのでなにを書いておいても動作には関係ありませんが，将来の互換性を考えてやはりrts命令が書いてあるアドレスを入れておいたほうがよいでしょう。

9番目は関数の名前を定義したテーブルの先頭アドレスを指します。関数名テーブルでは，それぞれの関数名の後ろに0を付け，次の関数名との区切りにしています。テーブルの最後はさらに0を置きます。つまり，関数名テーブルの終わりには0が2つ並ぶわけです。BASICから各関数を呼び

図1-A　外部関数のヘッダ部の構造

```
インフォメーションテーブル
dc.l  〈BASIC起動時の初期化ルーチンのアドレス〉
dc.l  〈RUN時に実行されるアドレス〉
dc.l  〈END時に実行されるアドレス〉
dc.l  〈SYSTEM/EXIT時に実行されるアドレス〉
dc.l  〈BREAK/ctrl-C入力時に実行されるアドレス〉
dc.l  〈1行入力中のctrl-D入力時に実行されるアドレス〉
dc.l  〈rts命令のあるアドレス(予備)〉
dc.l  〈rts命令のあるアドレス(予備)〉
dc.l  〈トークンテーブルの先頭アドレス〉
dc.l  〈パラメータテーブルの先頭アドレス〉
dc.l  〈実行アドレステーブルの先頭アドレス〉
          (以下20バイトは予備として0を入れておくこと)

トークンテーブル (PTR__TOKEN)
dc.b  'g_rev',0      ←関数名の終わりに0を付ける
dc.b  'peek',0
dc.b  0              ←テーブルの終わりにも0を付ける
          (関数名は64文字以内の英数字)

パラメータテーブル (PTR__PARAM)
dc.l  G_REV_PARAM    ←g_revのパラメータIDテーブルのアドレス
dc.l  PEEK_PARAM     ←peekのパラメータIDテーブルのアドレス

パラメータIDテーブル
G_REV_PARAM
    dc.w  Z_INT_OMT   ←g_revの引き数の型
    dc.w  Z_VOID_RET  ←g_revの戻り値の型
PEEK_PARAM
    dc.w  Z_INT_VAL
    dc.w  Z_INT_VAL
    dc.w  Z_INT_RET
          (パラメータIDの内容については図1-Bとリスト1-A参照)

実行アドレステーブル (PTR__EXEC)
dc.l  GRAPHIC_REVERSE ←g_revの実行開始アドレス
dc.l  MEMORY_READ     ←peekの実行開始アドレス
```

図1-B　パラメータIDの内容

ビット	意味
15	1：戻り値，0：引き数
14〜8	(未使用)
7	1：省略可能
6，5	0 0 → 単純変数 0 1 → 1次元配列 1 0 → 2次元配列 1 1 → 多次元配列
4	1：ポインタ，0：値
3	1：str
2	1：int
1	1：char
0	1：float

〈引き数〉

$0001	float_val	float型数値
$0002	int_val	int型数値
$0004	char_val	char型数値
$0008	str_val	文字列
$0011	float_vp	float型変数のデータ部へのポインタ
$0012	int_vp	int型変数のデータ部へのポインタ
$0014	char_vp	char型変数のデータ部へのポインタ
$0018	str_vp	文字型変数のデータ部へのポインタ
$0081	float_omt	省略可能なfloat型数値
$0082	int_omt	省略可能なint型数値
$0084	char_omt	省略可能なchar型数値
$0088	srt_omt	省略可能な文字列

〈戻り値〉

$8001	float_ret	float型数値
$8002	int_ret	int型数値
……		(char型は戻り値としては使えない)
$8008	str_ret	文字列
$FFFF	void_ret	戻り値なし

出すときにはここで定義した名前を使うことになります。

10番目は関数に与えるパラメータの型と順番を定義するためのものですが，ちょっと難しいのはこれが指す先にはパラメータの定義があるのではなく，「各関数のパラメータの定義してあるアドレスを並べたテーブルがある」ということです。C言語風にいうと，10番目は「パラメータのテーブルを指すポインタ配列へのポインタ」ということになります。つまり，これが指す先に書いてあるアドレスが最初の関数のパラメータを定義してあるところを示し，指す先＋4に書いてあるアドレスは次の関数のパラメータを定義してあるところを示しているのです。

図1の例ではまず，10番目はPTR_PARAMを指しています。PTR_PARAMのところには実際にパラメータを定義するテーブルを指すアドレスが，関数名を定義した順に並んでおり，最初はG_REV_PARAM，次はPEEK_PARAMをそれぞれ指しています。そしてG_REV_PARAM，PEEK_PARAMのところに目的のパラメータ定義テーブルがあるのです。

このようにワンクッション置くというのは一見よけいなことのように見えるかもしれませんが，じつはこのほうがアクセスする側，つまりBASICにとっては扱いが簡単なのです。パラメータの数は関数によって異なりますから，パラメータの定義をずらずらと並べても，どこからがどの関数のためのものか見つけるのに手間がかかります。その点，このようにワンクッション置けば，関数名を見つけて何番目の関数かがわかったら，それを4倍して（アドレスは4バイトで表現されます）PTR_PARAMに足し，そこに書いてあるアドレスを持ってくるだけですむのです。

実際の関数のパラメータの型は引き数の順に並び，最後は関数の戻り値を示すものになっています。引き数と関数の戻り値の区別は，たとえば引き数としてchar型を渡し，戻り値としてint型をとるようなものでは，最初に$0004（char 型の引き数であることを示す）が，次に$8002（int 型の戻り値であることを示す）がきます。戻り値を示す場合$8000以上，引き数の場合は$0100未満ですから，引き数の数を明示しなくても区別がつけられるのです。

11番目は関数の実行アドレスを並べたテーブルの先頭アドレスを指します。並べる順番は関数名を定義した順になります。図1の例ではg_revのプログラム本体はGRAPHIC_REVERSEというラベルのところから始まり，同様にpeekはMEMORY_READから始まるのです。

引き数の受け取りと値の返し方

ヘッダを正しく作っておけばBASICから予定の関数が呼び出せるようになります。それでは，たとえば

data＝peek(1,&HC00000)

などとしたとき，1や&HC00000といったパラメータはどのようにしてBASICから機械語ルーチンに引き渡されるのでしょうか。また，peek関数の戻り値（dataに代入される値）はどのようにして返せばよいのでしょうか。実際に関数を作ろうとしたときに問題となる，これらの点について調べてみましょう。

関数が呼び出されたときのスタックの状態は図2のようになります。スタックのいちばん上（A7＝SPの指すアドレス）には帰り先のアドレス（当然のことながら4バイト），その次（A7＋4）に引き数の総数が2バイトあり，以下は引き数が積まれています。引き数はひとつあたり10バイトで，積まれる順序はBASICプログラムで引き数を書く順です。たとえば，

jj(zillion, nicol, bigi)

とすれば，（A7＋6）にはzillionの値，（A7＋16）にはnicolの値，（A7＋26）にはbigiの値が入っています。

さてここで，ひと口に「値」といってもそれがchar（1バイト）型なのか，int（4バイト）型なのか，それともfloat（浮動小数点）型なのかといった，型の違いによって値の入り方は異なってきます。

Cコンパイラなどではchar型をint型に拡張するほかはほとんどこれといった細工はせず，そのままスタックに積み上げていますので，関数側では引き数の型を知ることはできません。このため，たとえば引き数としてint型を要求しているところにfloat型の値を渡しても呼ばれた関数はそんなことはいっさい関知せず，浮動小数点のデータをそのままint型のデータとして使用してしまうことになります。Cはプログラマの責任でなんでもできるのが売り物ですから，“小さな親切”的なものはいっさいやらないのが筋というものなのでこのようになるのでしょう。

これに対し，X-BASICはBASICと名を冠するだけにさすがにそこまで不親切ではありません。各10バイトは，先頭の2バイトが変数の型を示すために，残り8バイト（64ビット）が値を表現するのに使用されます。このため，どんな型の値がくるかわからないような場合でも，先頭の2バイトを判断することで残りの8バイトにどのようなフォーマットで値が入っているのかを判断しながら使うことができるようになります。

もっとも，外部関数の場合にはわざわざヘッダで引き数や戻り値の型を定義していることからもわかるように，型の管理はBASICが受け持ってくれます。型が違う場合には，可能なら適当に型変換をしてくれますし，できないならエラーメッセージを出して止まってしまい外部関数は呼び出されませんから，予期しないような型の引き数がくることはありません。あらかじめ与えられるパラメータが予測できるのですから簡単なものです。

あと問題なのは関数からの帰り方と値の返し方ですが，関数の戻り値はひとつしかありませんので単純なものです。まず，関数が正常に終了したかどうかをBASICインタプリタに知らせるためにD0を使います。処理が正常に終了したらD0に0を入れておきます。引き数の値が規定値になっていないなど関数側でエラーと判定した場合にはD0に0以外の値を入れ，A1にエラーメッセージの文字列（文字列の終わりは0で示す）のあるアドレスを入れてリターンします。BASIC側ではD0に0以外の値が入っているとエラーメッセージを出力して実行を中断

図2　関数が呼び出されたときのスタック

※引き数の型

0	float型
1	int型（値は下位を使用）
2	char型（値は下位を使用。そのうちの最下位バイトが有効で，上位3バイトには0が入る）
3	str型
$FFFF	省略された引き数

します。

関数の戻り値は引き数のときと同じく10バイトのエリアを使って表現します。このエリアの先頭アドレスをA0に入れてリターンすれば，BASIC側で受け取ってもらえます。値を返さない関数，たとえば画面消去なども当然存在するわけで，このときはA0は無視されますから値はなんでもかまいません。

ちょっと便利なユーザー関数

なるべくやさしくしようと思っていろいろ書いてみましたが，さすがに関数という形をとると，これまでのようなBASICからコールするだけの機械語サブルーチンと違い，いろいろな条件が付いてきてわかりにくいものです。なにかサンプルプログラムがあったほうがいいかな，ということで，ちょっと便利そうな関数を作ってみました（リスト1）。

グラフィック画面（512×512ドットモード）の色反転を行うg_rev( )，左右反転を行うg_inv( )，IOCSコールによる画面モードの切り換えを行うg_mode(n)，メモリの内容を読み出すpeek(size, adrs)，メモリへの書き込みを行うpoke(size, adrs, data)の5つです。

peek, pokeの最初のパラメータはアクセスサイズを意味し，0でバイトアクセス，1でワードアクセス，2でロングワードアクセスになります。8086CPUならバイトアクセスとワードアクセスしかありませんから，pokeb, pokewのようにアクセスサイズごとに別々の関数を用意してもよいのでしょうが，68000ではこれにロングワードのアクセスが加わります。peek, pokeだけで6つも関数を作るとサンプルプログラムがヘッダだらけになりそうだということもあって，ひとつにまとめて引数でアクセスサイズを指定するようにしてみました。

g_mode( )は画面モード設定のIOCSコールをそのまま利用しました。パラメータの値のチェックはしていませんので，IOCSコール一覧にない値でもそのままIOCSを呼び出します。このとき画面がどうなるかは保証されていませんが，IOCSの飛び先を追いかけてみたら，なにがしか意味のある動きをする場合もありそうなコーディングでした（使えるかどうかは別にして）。

さて，これらをBASICの外部関数として登録してみましょう。ソースファイル名はusrfnc.sとしておきます。これをアセンブル，リンクします（includeファイルはあらかじめ用意しておいてください。これは先月号とまったく同じものです）。

as b：usrfnc

lk b：usrfnc－o usrfnc. fnc

できあがったファイル，usrfnc.fncをシステムディスク（もちろんいつも使っているコピーしたもの）のBASICディレクトリの下にコピーします。

copy b：usrfnc.fnc a：¥basic

そしてエディタなどでBASIC.CNFのファイルの中に，FUNC＝usrfncの1行を追加します。

さあBASICを起動しましょう。5つの新しい関数が追加されたBASICのできあがりです。試しにg_rev( )とg_inv( )，g_mode(n)を使ったグラフィックプログラムを作ってみましたので試してみてください（サンプル1）。

サンプル1　usrfncを使った例

```
1000 /*------ User Function Sample Program -------*/
1010 screen 1,3,1,1
1020 g_mode(&H10C):wipe():console 0,31,0
1030 int i,j,k,l
1040 for l=1 to 4
1050    for k=0 to 10
1060       for j=0 to 12
1070          for i=0 to 23
1080             circle(k*l*10+50,j*35+40,i,hsv(j*10+k,31,31-i))
1090          next
1100       next
1110    next
1120    user()
1130 next
1140 end
1150 /*--------------------------------------*/
1160 func user()
1170    g_inv():delay()
1180    g_inv():delay()
1190    g_inv():delay()
1200    g_rev():delay()
1210 endfunc
1220 /*--------------------------------------*/
1230 func delay()
1240   int i
1250   for i=0 to 5000:next
1260 endfunc
```

オマケのヘッダジェネレータ

関数を作る過程の中で，やはり面倒くさいのがヘッダの作成です。X-BASICの外部関数のヘッダはBASICの起動時や，RUN, END, SYSTEMなどの実行時にも実行されるプログラムを指定できるなど，なかなかマニアには受けそうな構造をしているのですが，それだけにどうしても指定しなければならないことが多くなり，ちょっとしたプログラムを外部関数として登録したいというような場合にはよけいなことまで面倒をみさせられているようで煩わしいことこの上ありません。

私自身，先ほどのサンプルプログラムを書いていて，ヘッダの作成だけでうんざりしてしまいました。せめてテーブルのひな形ができていて，アドレスだけあとで変更する程度になっていればずっと簡単なのに。オールドの黒い瓶に映ったVT-62のフィラメントを眺めていて頭に浮かんだのが「面倒な作業はコンピュータにやらせてしまえ」という考え。ロン・カーターの「ダブルベ

サンプル2　ヘッダジェネレート例

A）定義ファイル

```
func_test ( i;int, j;char);float
func_test1()
func_test2(i)
```

B）A)をもとに作成したヘッダファイル

```
        .include  osmac.inc
        .include  iocsmac.inc
        .text
        dc.l            X_INZ
        dc.l            X_RUN
        dc.l            X_END
        dc.l            X_SYSTEM
        dc.l            X_BRK
        dc.l            X_CTRL_D
        dc.l            X_RETADRS1
        dc.l            X_RETADRS2
        dc.l            PTR_TOKEN
        dc.l            PTR_PARAM
        dc.l            PTR_EXEC
        dc.l            0,0,0,0,0
PTR_TOKEN:
        dc.b            'func_test',EOS
        dc.b            'func_test1',EOS
        dc.b            'func_test2',EOS
        dc.b            EOS
PTR_PARAM:
        dc.l            func_test_param
        dc.l            func_test1_param
        dc.l            func_test2_param
func_test_param:
        dc.w            Z_INT_VAL
        dc.w            Z_CHAR_VAL
        dc.w            Z_FLOAT_RET
func_test1_param:
        dc.w            Z_VOID_RET
func_test2_param:
        dc.w            Z_INT_VAL
        dc.w            Z_VOID_RET
PTR_EXEC:
        dc.l            func_test
        dc.l            func_test1
        dc.l            func_test2
X_INZ:
X_RUN:
X_END:
X_SYSTEM:
X_BRK:
X_CTRL_D:
X_RETADRS1:
X_RETADRS2:
PTR_TOKEN:
PTR_PARAM:
PTR_EXEC:
                rts
```

ース」とともにキーボードを叩き，作ってみたのが名付けて「ヘッダジェネレータ」。関数の名前と引き数，戻り値の型などをふつうのBASICでのfunc文のように書いておけば，それを参考にしてヘッダを作成してくれるプログラムです（リスト2）。

まず，関数名などを登録したファイルを作っておきます。ファイル名はtest. namのように拡張子を「.nam」にしてください。この中に，

```
sample(i ; int, j ; char) ; int
non(no ; str, an)
    :
    :
```

というように，BASICのfunc文の後ろにくるような形で関数の名前を書いておきます。変数の名前は無視されますのでなんでもかまいません。変数の型を省略した場合には(nonのanのように）int 型であると判断されます。関数自体の型を付ければその型が戻り値に指定され，省略すれば戻り値なし(Cでいうvoid型）です。

できあがったらヘッダジェネレータを動かします。最初にファイル名を聞いてきますので「.nam」を取った名前（test. namならtest）を入力してください。すると，ゾロゾロっとメッセージが出て，ヘッダを作り始めます。作成したヘッダのファイル名は先ほどのファイル名の拡張子を「.inc」にしたもの（testならtest. inc）になっています。あとはこのヘッダをincludeファイルとして取り込むか，エディタで自分の作ったプログラムとくっつけるようにすればずっと簡単に外部関数を作成することができるでしょう。

ゲームも言語も関数になる

X-BASIC の外部関数を実際に作ってみると，あれもできるこれもできる，とどんどん新しい発想が生まれてきます。関数が増えるという考えも，次々と新しい機械語サブルーチンを追加できるという考えもよいでしょう。もっと極端に考えれば，単にヘッダを付けて最後にrts命令を実行するようにしておけば，あらゆるプログラムをBASICから簡単に扱えるということです。

グラディウスをロードして実行する外部関数を作り，ゲームの終了でrtsするようにしておけば，BASICで

1000 gradius( )

というように書くとグラディウスが走って，全部やられると帰ってくるようにもできるのです。つまり，gradius( )は「ゲームをやって終了したら帰ってくる」関数というわけです（スコアを関数の値として返せたら楽しいですね)。

もちろん，走るのは他の言語処理系のプログラムでもいいわけで，文字配列にプログラムを入れて呼び出すと，その中身を読んで実行するなりコンパイルするなりして帰ってくる，たとえばfortran( )やapl( )，lisp( )，prolog( )なんていう関数を作ってもいっこうにかまわないのです。ある目的のためだけに作られた言語をBASICから呼び出すようにしてそれぞれの言語の得意分野を生かす，サブインタプリタといったような考え方も面白いと思いませんか？ まさに「なんでもできちゃうツインタワー」の「天下無敵のBASIC」ですね。

リスト1-A　osmac.incソースリスト

```
*----------------------------------------
*-          Definishion file            -
*----------------------------------------
.nlist
*
*...... OS system call...........
*

_EXIT           equ     $ff00
_GETCHAR        equ     $ff01
_PUTCHAR        equ     $ff02
_COMINP         equ     $ff03
_COMOUT         equ     $ff04
_PRNOUT         equ     $ff05
_INPOUT         equ     $ff06
_INKEY          equ     $ff07
_GETC           equ     $ff08
_PRINT          equ     $ff09
_GETS           equ     $ff0a
_KEYSNS         equ     $ff0b
_KFLUSH         equ     $ff0c
_FFLUSH         equ     $ff0d
_CHGDRV         equ     $ff0e
_DRVCTRL        equ     $ff0f
_CONSNS         equ     $ff10
_PRNSNS         equ     $ff11
_CINSNS         equ     $ff12
_COUTSNS        equ     $ff13
_CURDRV         equ     $ff19
_GETSS          equ     $ff1a
_SUPER          equ     $ff20
_FNKKEY         equ     $ff21
_KNJCTRL        equ     $ff22
_CONCRTRL       equ     $ff23
_KEYCTRL        equ     $ff24

*
* BASIC constants
*
Z_FLOAT_VAL     equ     $0001
Z_INT_VAL       equ     $0002
Z_CHAR_VAL      equ     $0004
Z_STR_VAL       equ     $0008
Z_FLOAT_VP      equ     $0011
Z_INT_VP        equ     $0012
Z_CHAR_VP       equ     $0014
Z_STR_VP        equ     $0018
Z_FLOAT_OMT     equ     $0081
Z_INT_OMT       equ     $0082
Z_CHAR_OMT      equ     $0084
Z_STR_OMT       equ     $0088
Z_ARY1_FIC      equ     $0037
Z_ARYA_C        equ     $0034
Z_FLOAT_RET     equ     $8000
Z_INT_RET       equ     $8001
Z_STR_RET       equ     $8003
Z_VOID_RET      equ     $ffff

*
*...... System constants .......
*
VRAM            equ     $c00000

*
*...... Generall constants........
*
READY           equ     0
ERROR           equ     1
CR              equ     $0d
LF              equ     $0a
EOS             equ     0
.list
```

リスト1-B　usrfncアセンブルリスト

```
 1 00000000                            *
 2 00000000                            *
 3 00000000                            * User Function sample program
 4 00000000                            *
 5 00000000                            *
 6 00000000                                        .include        osmac.inc
 6 00000000                            *----------------------------------
 6 00000000                            *-          Definishion file      -
 6 00000000                            *----------------------------------
 6 00000000                            .list
 7 00000000
 8 00000000                                        .text
 9 00000000 (01)0000009C                           dc.l            X_INZ
10 00000004 (01)0000009C                           dc.l            X_RUN
11 00000008 (01)0000009C                           dc.l            X_END
12 0000000C (01)0000009C                           dc.l            X_SYSTEM
13 00000010 (01)0000009C                           dc.l            X_BRK
14 00000014 (01)0000009C                           dc.l            X_CTRL_D
15 00000018 (01)0000009C                           dc.l            X_RETADRS       * spare
16 0000001C (01)0000009C                           dc.l            X_RETADRS       * spare
17 00000020 (01)00000040                           dc.l            PTR_TOKEN
18 00000024 (01)0000005E                           dc.l            PTR_PARAM
19 00000028 (01)00000088                           dc.l            PTR_EXEC
20 0000002C 0000000000000000                       dc.l            0,0,0,0,0
            0000000000000000
            00000000
21 00000040
22 00000040                            *.......... Function name table...........
23 00000040 675F72657600               PTR_TOKEN   dc.b            'g_rev',EOS
24 00000046 675F696E7600                           dc.b            'g_inv',EOS
25 0000004C 675F6D6F646500                         dc.b            'g_mode',EOS
26 00000053 7065656B00                             dc.b            'peek',EOS
27 00000058 706F6B6500                             dc.b            'poke',EOS
28 0000005D 00                                     dc.b            EOS
29 0000005E                            *....... Function Parameters table .......
30 0000005E (01)00000072               PTR_PARAM   dc.l            G_REV_PARAM
31 00000062 (01)00000074                           dc.l            G_INV_PARAM
32 00000066 (01)00000076                           dc.l            G_MODE_PARAM
33 0000006A (01)0000007A                           dc.l            PEEK_PARAM
```

```
 34 0000006E (01)00000080                    dc.l        POKE_PARAM
 35 00000072
 36 00000072 FFFF            G_REV_PARAM     dc.w        Z_VOID_RET      * g_rev()
 37 00000074 FFFF            G_INV_PARAM     dc.w        Z_VOID_RET      * g_inv()
 38 00000076 0002            G_MODE_PARAM    dc.w        Z_INT_VAL       * g_mode(n)
 39 00000078 FFFF                            dc.w        Z_VOID_RET
 40 0000007A 0002            PEEK_PARAM      dc.w        Z_INT_VAL       * peek(size,adrs)
 41 0000007C 0002                            dc.w        Z_INT_VAL
 42 0000007E 8001                            dc.w        Z_INT_RET
 43 00000080 0002            POKE_PARAM      dc.w        Z_INT_VAL       * poke(size,adrs,data)
 44 00000082 0002                            dc.w        Z_INT_VAL
 45 00000084 0002                            dc.w        Z_INT_VAL
 46 00000086 FFFF                            dc.w        Z_VOID_RET
 47 00000088                 *........ Function address table .........
 48 00000088 (01)0000009E    PTR_EXEC        dc.l        GRAPHIC_REVERSE
 49 0000008C (01)000000C6                    dc.l        GRAPHIC_INVERT
 50 00000090 (01)000000FE                    dc.l        GRAPHIC_MODE
 51 00000094 (01)00000110                    dc.l        MEMORY_READ
 52 00000098 (01)00000154                    dc.l        MEMORY_WRITE
 53 0000009C
 54 0000009C                 X_INZ:
 55 0000009C                 X_RUN:
 56 0000009C                 X_END:
 57 0000009C                 X_SYSTEM:
 58 0000009C                 X_BRK:
 59 0000009C                 X_CTRL_D:
 60 0000009C 4E75            X_RETADRS       rts
 61 0000009E                 *........ Reverse graphic screen .......
 62 0000009E                 GRAPHIC_REVERSE:
 63 0000009E 42A7                            clr.l       -(sp)
 64 000000A0 FF20                            dc.w        _SUPER
 65 000000A2 588F                            addq.l      #4,sp
 66 000000A4 207C00C00000                    movea.l     #VRAM,a0
 67 000000AA 303C0003                        move.w      #3,d0
 68 000000AE                 GRREV:
 69 000000AE 323CFFFF                        move.w      #-1,d1
 70 000000B2 4658            GRREV64K        not.w       (a0)+
 71 000000B4 51C9FFFC_000000B2               dbra        d1,GRREV64K
 72 000000B8 51C8FFF4_000000AE               dbra        d0,GRREV
 73 000000BC 7000                            move.l      #READY,d0
 74 000000BE 41F9(01)000001C6                lea.l       RETDAT,a0
 75 000000C4 4E75                            rts
 76 000000C6                 *........ Invert graphic screen ......
 77 000000C6                 GRAPHIC_INVERT:
 78 000000C6 42A7                            clr.l       -(sp)
 79 000000C8 FF20                            dc.w        _SUPER
 80 000000CA 588F                            addq.l      #4,sp
 81 000000CC 207C00C00000                    movea.l     #VRAM,a0
 82 000000D2 203C000001FF                    move.l      #511,d0         * Line counter
 83 000000D8                 INV_SCRN:
 84 000000D8 2248                            movea.l     a0,a1
 85 000000DA D1FC00000400                    adda.l      #1024,a0
 86 000000E0 2448                            movea.l     a0,a2
 87 000000E2 323C00FF                        move.w      #255,d1
 88 000000E6
 89 000000E6 3411            INV_1_LINE      move.w      (a1),d2
 90 000000E8 32E2                            move.w      -(a2),(a1)+
 91 000000EA 3482                            move.w      d2,(a2)
 92 000000EC 51C9FFF8_000000E6               dbra        d1,INV_1_LINE
 93 000000F0 51C8FFE6_000000D8               dbra        d0,INV_SCRN
 94 000000F4 7000                            move.l      #READY,d0
 95 000000F6 41F9(01)000001C6                lea.l       RETDAT,a0
 96 000000FC 4E75                            rts
 97 000000FE                 *...... Set graphic screen mode .......
 98 000000FE                 GRAPHIC_MODE:
 99 000000FE 222F000C                        move.l      12(sp),d1
100 00000102 7010                            move.l      #$10,d0
101 00000104 4E4F                            trap        #$0f
102 00000106 7000                            move.l      #READY,d0
103 00000108 41F9(01)000001C6                lea.l       RETDAT,a0
104 0000010E 4E75                            rts
105 00000110                 *............ Memory read ............
106 00000110                 MEMORY_READ:
107 00000110 42A7                            clr.l       -(sp)           * Enter supervisor state
108 00000112 FF20                            dc.w        _SUPER
109 00000114 588F                            addq.l      #4,sp
110 00000116 202F000C                        move.l      12(sp),d0       * Access size check
111 0000011A 0C8000000002                    cmpi.l      #2,d0           * Access size <= 2 ?
112 00000120 6F08_0000012A                   ble         RD_MODE_OK      * Yes, then branch
113 00000122 43F9(01)0000019E                lea.l       E_MODE,a1       * Error message set
114 00000128 606A_00000194                   bra         ERROR_AUX
115 0000012A 4281            RD_MODE_OK      clr.l       d1
116 0000012C 242F0016                        move.l      22(sp),d2       * get address
117 00000130 2642                            movea.l     d2,a3
118 00000132 51C80006_0000013A               dbra        d0,RD_NOTBYTE   * D0 == 0 ?
119 00000136 1213                            move.b      (a3),d1         * yes, then peek memory (BYTE)
120 00000138 600A_00000144                   bra         RD_READY_AUX
121 0000013A 51C80006_00000142 RD_NOTBYTE    dbra        d0,RD_NOTWORD   * D0 == 1 ?
122 0000013E 3213                            move.w      (a3),d1         * yes, then peek memroy (WORD)
123 00000140 6002_00000144                   bra         RD_READY_AUX
124 00000142 2213            RD_NOTWORD      move.l      (a3),d1         * LONG WORD Access
125 00000144 23C1(01)000001CC RD_READY_AUX   move.l      d1,INT_DATA     * store return data
126 0000014A 41F9(01)000001C6                lea.l       RETDAT,a0       * return data pointer setup
127 00000150 7000                            move.l      #READY,d0       * return status = READY
128 00000152 4E75                            rts                         * Bye,bye
129 00000154
130 00000154                 *............ Memory write ............
131 00000154                 MEMORY_WRITE:
132 00000154 42A7                            clr.l       -(sp)           * Enter supervisor state
133 00000156 FF20                            dc.w        _SUPER
134 00000158 588F                            addq.l      #4,sp
135 0000015A 202F000C                        move.l      12(sp),d0       * Access size check
136 0000015E 0C8000000002                    cmpi.l      #2,d0           * Access size <= 2 ?
137 00000164 6F08_0000016E                   ble         WR_MODE_OK      * Yes, then branch
138 00000166 43F9(01)0000019E                lea.l       E_MODE,a1       * Error message set
139 0000016C 6026_00000194                   bra         ERROR_AUX
140 0000016E 242F0016        WR_MODE_OK      move.l      22(sp),d2       * get address
141 00000172 2642                            movea.l     d2,a3
142 00000174 222F0020                        move.l      32(sp),d1
143 00000178 51C80006_00000180               dbra        d0,WR_NOTBYTE   * D0 == 0 ?
144 0000017C 1681                            move.b      d1,(a3)         * yes, then poke memory (BYTE)
145 0000017E 600A_0000018A                   bra         WR_READY_AUX
146 00000180 51C80006_00000188 WR_NOTBYTE    dbra        d0,WR_NOTWORD   * D0 == 1 ?
147 00000184 3681                            move.w      d1,(a3)         * yes, then poke memroy (WORD)
148 00000186 6002_0000018A                   bra         WR_READY_AUX
149 00000188 2681            WR_NOTWORD      move.l      d1,(a3)         * LONG WORD Access
150 0000018A 41F9(01)000001C6 WR_READY_AUX   lea.l       RETDAT,a0       * return data pointer setup
151 00000190 7000                            move.l      #READY,d0       * return status = READY
152 00000192 4E75                            rts                         * Bye,bye
153 00000194
154 00000194                 *............ Error Return ............
155 00000194 41F9(01)000001C6 ERROR_AUX      lea.l       RETDAT,a0       * return data pointer setup (Nothing to mean)
156 0000019A 7001                            move.l      #ERROR,d0       * return status = ERROR
157 0000019C 4E75                            rts
158 0000019E
```

```
159 0000019E 0D0A8341834E835A  E_MODE       dc.b         CR,LF,'アクセス・モードは0から2までです。',EOS
             835881458382815B
             836882CD824F82A9
             82E7825182DC82C5
             82C582B7814200
160 000001C5 000000            RETDAT       dc.w         0
161 000001C8 00000000                       dc.l         0
162 000001CC 00000000          INT_DATA     dc.l         0
163 000001D0                                end
```

リスト2 ヘッダジェネレータ

```
1000 /*--------- User function headder table genereter -----------*/
1010 str  CRLF,CTAB    /* 'tab' is reserved  */
1020 CRLF=chr$(&HD)+chr$(&HA):CTAB=chr$(&H9)
1030 dim str  fname(20)
1040 dim str  vname[80]
1050 dim char vtype(20,10)
1060 dim char ftype(20)
1070 str rfilename
1080 str s[80],t[80]
1090 str typetable(5) = {
1100          "float",
1110          "int",
1120          "char",
1130          "str",
1140          "void"  }
1150 str paramtype(5) = {
1160          "Z_FLOAT_VAL",
1170          "Z_INT_VAL",
1180          "Z_CHAR_VAL",
1190          "Z_STR_VAL",
1200          "Z_INT_OMT" }
1210 str functype(5) = {
1220          "Z_FLOAT_RET",
1230          "Z_INT_RET",
1240          "Z_CHAR_RET",
1250          "Z_STR_RET",
1260          "Z_VOID_RET" }
1270 int n,textp
1280 int rfp,wfp
1290 input "Function table file name input ";rfilename
1300 rfp = fopen(rfilename+".nam","r")
1310 if rfp = -1 then print"File not found!":end
1320 wfp = fopen(rfilename+".inc","c")
1330 if wfp = -1 then print"Can't create output file!":end
1340 n=0
1350 print"Function file read"
1360 repeat
1370          if freads(s,rfp)=-1 then {
1380                  generate()
1390                  end
1400          }
1410          print:print s
1420          textp=0
1430          vn=0
1440          if gettoken()=0 then end
1450          fname(n)=t
1460          if gettoken()<>'(' then erraux()
1470          gettoken()
1480          while  t[0]<>')'
1490                  if not isalpha(t[0]) then erraux()
1500                  vname = t:print"<";t;":";
1510                  switch gettoken()
1520                          case ';': gettoken()
1530                                    print t;">";
1540                                    setvtype(t)
1550                                    if gettoken()=',' then gettoken()
1560                                    break
1570                          case ',': print "INT";">";
1580                                    setvtype("int")
1590                                    gettoken()
1600                                    break
1610                          case ')': print "INT";">";
1620                                    setvtype("int")
1630                                    break
1640                          default:  break
1650                  endswitch
1660          endwhile
1670          vtype(n,0)=vn
1680          if gettoken()=';' then {
1690                          gettoken()
1700                  print t
1710                  setftype(t)
1720          } else {
1730                  print "VOID"
1740                  setftype("void")
1750          }
1760 until 0
1770 end
1780 /*
1790 func gettoken()
1800    int  i
1810    char c
1820    for i=0 to 79
1830            t[i]=0
1840    next
1850    skipspace()
1860    i=1
1870    c=s[textp]
1880    t[0]=c
1890    textp=textp+1
1900    if isnamchr(c) then {
1910            c=s[textp]
1920            while isnamchr(c)
1930                    t[i]=c
1940                    i=i+1
1950                    textp=textp+1
1960                    c=s[textp]
1970            endwhile
1980            skipspace()
1990    }
2000    return(t[0])
2010 endfunc
2020 /*
2030 func skipspace()
2040   while isspace(s[textp])
2050            textp=textp+1
2060   endwhile
2070 endfunc
2080 /*
2090 func erraux()
2100   print "ERROR":end
2110 endfunc
2120 /*
2130 func isnamchr(c;char)
2140   if isalnum(c) or (c='_') then return (-1) else return(0)
2150 endfunc
2160 /*
2170 func setvtype(s;str)
2180   int i
2190   vn=vn+1
2200   for i=0 to 3
2210          if s=typetable(i) then {
2220                  vtype(n,vn)=i
2230                  return()
2240          }
2250   next
2260   erraux()
2270 }
2280 /*
2290 func setftype(s;str)
2300   int i
2310   for i=0 to 4
2320          if s=typetable(i) then {
2330                  ftype(n)=i
2340                  n=n+1
2350                  return()
2360          }
2370   next
2380   erraux()
2390 }
2400 /*
2410 func generate()
2420    int i,j,vn
2430    print"Standard header table generate ......... ";
2440    fwrites(CTAB+".include  osmac.inc"+CRLF,wfp)
2450    fwrites(CTAB+".include  iocsmac.inc"+CRLF,wfp)
2460    fwrites(CTAB+".text"+CRLF,wfp)
2470    dclwrite('l',"X_INZ")
2480    dclwrite('l',"X_RUN")
2490    dclwrite('l',"X_END")
2500    dclwrite('l',"X_SYSTEM")
2510    dclwrite('l',"X_BRK")
2520    dclwrite('l',"X_CTRL_D")
2530    dclwrite('l',"X_RETADRS1")
2540    dclwrite('l',"X_RETADRS2")
2550    dclwrite('l',"PTR_TOKEN")
2560    dclwrite('l',"PTR_PARAM")
2570    dclwrite('l',"PTR_EXEC")
2580    dclwrite('l',"0,0,0,0,0")
2590    print"Done"
2600    print"Function name table generate .......... ";
2610    lblwrite("PTR_TOKEN")
2620    for i=0 to n-1
2630            dclwrite('b',"'"+fname(i)+"',EOS")
2640    next
2650    dclwrite('b',"EOS")
2660    print"Done"
2670    print"Parameter pointer table generate ....... ";
2680    lblwrite("PTR_PARAM")
2690    for i=0 to n-1
2700            dclwrite('l',fname(i)+"_param")
2710    next
2720    print"Done"
2730    print"Parameter table generate  ............. ";
2740    for i=0 to n-1
2750            vn = vtype(i,0)
2760            lblwrite(fname(i)+"_param")
2770            for j=1 to vn
2780                    dclwrite('w',paramtype(vtype(i,j)))
2790            next
2800            dclwrite('w',functype(ftype(i)))
2810   next
2820   print"Done"
2830   print"Exec address table generate ........... ";
2840   lblwrite("PTR_EXEC")
2850   for i=0 to n-1
2860            dclwrite('l',fname(i))
2870   next
2880   print"Done"
2890   print"Standard exec address table generate ... ";
2900   lblwrite("X_INZ")
2910   lblwrite("X_RUN")
2920   lblwrite("X_END")
2930   lblwrite("X_SYSTEM")
2940   lblwrite("X_BRK")
2950   lblwrite("X_CTRL_D")
2960   lblwrite("X_RETADRS1")
2970   lblwrite("X_RETADRS2")
2980   lblwrite("PTR_TOKEN")
2990   lblwrite("PTR_PARAM")
3000   lblwrite("PTR_EXEC")
3010   fwrites(CTAB+CTAB+"rts"+CRLF,wfp)
3020   print"Done"
3030 endfunc
3040 /*
3050 func dclwrite(c;char,s;str)
3060          fputc(&H9,wfp)
3070          fwrites("dc.",wfp)
3080          fputc(c,wfp)
3090          fputc(&H9,wfp):fputc(&H9,wfp)
3100          fwrites(s+CRLF,wfp)
3110 endfunc
3120 /*
3130 func lblwrite(s;str)
3140          fwrites(s+":"+CRLF,wfp)
3150 endfunc
```

特集

迷宮の日本語処理環境

日本語ワードプロセッサ，この言葉が定着するようになってからまだそう長い年月は経過していません。しかし，その間にもワープロの需要は拡大し，業務ではもちろんのこと大学の論文などもワープロ原稿で，といったことが増えてきているようです。
そしてワープロと接する機会が多くなってきた現状と同時に，さまざまなタイプのワープロソフトや専用機が登場してきています。前回ワープロ特集を組んだとき，ポータブルワープロを例にとって考えてみると辞書の語数の多さだけが売りもので，ディスプレイは1行表示，ディスクも装備されていませんでした。あれから約2年，ワープロソフトは付属機能が充実し，ワープロ専用機はますますパソコンと競合できるマシンとして成長してきています。しかしワープロが使われる状況というのは千差万別のはず，使う機会が増えれば増えるほど非常に個人的な欲求を満足してくれる道具としての充実が望まれます。
そういった環境の変化のなかで，私たちが満足して使いたいワープロというものには複雑な日本語表現を適切に処理し，さらには想像力をも高めてくれる，よりいっそう身近な道具としての期待が高まってきます。清書マシンから思考ツールへ，ワープロ次世代へのメッセージをこれから皆さんといっしょに考えてみようではありませんか。

特集 迷宮の日本語処理環境

使うのは私なのである

Iwai Ippei
祝 一平

祝一平怒る。すべての日本語ワープロにとって裁きの時が訪れた。天はタコのうえにワープロを作らず，タコのしたにもワープロを作らない。清く正しいエディタなしに日本語は扱えないのである。

タコぢゃ，タコぢゃ，タコワープロぢゃ！

結論から先にいわせてもらうと，現在パソコン用として発売されている日本語ワープロはほとんどタコである。はっきりいっていま私がこの原稿を書いているX1用のワープロもタコである（タコではあっても，ほかのワープロよりははるかにましなタコなので使っているのだが)。

ここではワープロといったが，はっきりいって現在の日本語ワープロの問題点は変換機能ではなく，「エディタとしてタコだ」ということに尽きるのである。すなわち肝心の根本が腐っているのだ（もちろんほかの部分も腐っている)。そしてほとんど信じられないことだが，ワープロのメーカーでそのことに気が付いているらしいところはなく，「新方式の変換機能」とか「変換速度の向上」とか「超多機能」とかに血道をあげているのである。

そこでここでは，悪い機能は元から断たなきゃだめということで，「タコエディタとしての日本語ワープロ」を中心にして書いてみるのであった。

恐怖のキーバッファ

初めて見たときにはほとんどひっくり返ってしまったのであるが，まったくもって信じられないのがバックスペース（DELキー）がキーバッファにたまるというやつである。

たとえば，たいていのBASICのエディタでは，DELキーを押し続けるときちんとキーリピートがかかり「てててて」と文字が削除されていくのである。そして「ここだ」というところまで削除されたときに指を離すと，ピタリと止まってくれるのである。そこで普通の人間はそれと同じ感覚でワープロを使うことになる。そして，自分の指が離されたにもかかわらず，相変わらず文字を削除し続けるカーソルを見て唖然とするのである。私は断言してはばからないのであるが，そのようなワープロを作るようなヤローはサギ師なのである。とりあえずはワープロの形をしたプログラムを書けるだけの技量があるのだから，少し考えればそんなことは避けることができたはずなのである。

そもそも削除に関するキー操作に対しては，特別な配慮が必要なのだ。それさえもわからないようなヤローにプログラマを名のる資格はないのである。しかし，実際にはそのようなワープロが販売されていたりする（当然のことながら，ほとんど売れていないらしいが)。

なぜファンクションキーなのか

日本語ワープロを使っていて気になってしようがないことのひとつに，「ファンクションキーを多用しなければならない」ということがある。少し考えればわかることだと思うが，本当はファンクションキーに機能を盛り込むことは避けるべきなのだ。理由はまず第一に「ホームポジションから遠いのでブラインドタッチしづらい」ということである。恐らくブラインドタッチでファンクションキーを押せるような人はほとんどいないであろう。そして第二点としては，結局は同じことなのであるが，「指がホームポジションから離れる」ということである。いうまでもないことだとは思うが，念のためにいっておく。

頻繁に指をホームポジションから外さなければいけなかったりすると，ものすごくイライラしてくるものなのだよ。

たとえば，私がいま使っているワープロでは，恐ろしいことにファンクションキーにA～Eまでのモードがあって，[F1]によってそのモードがA→B→C→D→E→Aとサイクリックに変わることになっている。これは実に最低で，たとえば文書の一部をほかの場所に複写するには，[F1][F1][F3]とすることになる（その後範囲の指定→複写先の指定を行う)。

この操作を聞いて，たいていの人は「なんだ[F1]を2回押さなければならない以外は普通の操作じゃないか」と思う人がほとんどであろう。

悲しいことである。

日本にはまともなエディタ文化がないからこんなことになってしまうのである。

確かにこれでも機能は果たしているのである。しかし「機能を果たしている」ということと，「よいものであるか」どうかとは別なのである。

ひょっとしたらわかっていないかもしれないので念のためにいっておくが，複写とか移動とか削除とかは「ものすごく基本的な操作」なのである。基本的な操作である以上は，そのキー操作はすんなりとできるものでなくてはならないのだ。「ファンクションキーを押す」ということは確かに単純でわかりやすいことではあるが，誰がなんといっても**すんなり**ではないのである。困ったことにワープロメーカーはこんな簡単なことさえもわかっていないのである。なにせローマ字入力が絶対不可欠な必須アイテムである，ということを理解するまでにあれだけの時間がかかった程度の人々なのだ。

というわけで，ワープロのメーカーはファンクションキー主義を改めるつもりはまったくないようである。これは自分でやってみればわかるはずである。ファンクションキーに指を伸ばすのは一般の打鍵動作としては不自然なのだ。だから印刷とか，書式の設定とかのたまにしか使わない機能に割り当てておくべきだ。まかり間違っても編集コマンドに割り当ててはいけないのだ。

これに関連したことに，[変換キー]の問題がある。一般的に変換キーは[XFER]なのであるが，どっこいX1 turboの場合はこのキーはキーボードの右下にあるのである。よってとーぜんのことながら[XFER]

キーを押すたびに手がホームポジションから離れてしまう。これは実に最低である。

この点に関しては，たとえば98用に一番売れているワープロ「一太郎」ではどうなっているかというと，確定していないときは「スペース」が変換キーになるようになっている。確かにそれならば手をホームポジションから動かさなくてすむが，やはり，ちょっと問題があるのではないかという気がする。

しかし解決するのは簡単である。すなわちコントロールコードのひとつにも変換キーの機能を割り振ってしまえばよいのである。そうすれば指をホームポジションから動かさずにすむのである。どーだ，実に単純な解決法だろう。しかしワープロメーカーはそんなことを思いつきもしないのである（ただし，面と向かって聞いたなら「考慮はしてみたんですがねー」とかいうに決まっているのだろーが）。

この「キーの割り当て」に関しては山ほど文句がある。たとえば，私のようにアルファベットがしょっちゅう出てくる文章を書いている人間にとっては，モード切り換えのキーの位置はものすごく重要なことなのである。また，このことはたいていのワープロでは解決されていることなのだと思うが，それでもなかには（つまりいま私が使っているワープロなのだが），コントロールキーにカーソル移動キーが割り当てられていないものもあるのである。当たり前すぎていうまでもないことだが，カーソルキーを使えば指はホームポジションから外れてしまうのである。

本当に彼らはなにを考えて生きているのだろうか？

勝手に決めるな

また私がこのワープロに対して持っている不満のひとつに「キー入力時のクリック音がない」ということがある。実はこの不満について話し合っているときに，スタッフの間から「クリック音は耳障りだから付けなかったのでしょ」という発言があった。確かに一理ある。おそらくワープロの設計者もそう考えて，クリック音を付けなかったのであろう。しかしここで考えてほしいのは，

「クリック音がうるさいかどうかを決めるのは，メーカーではなく**ユーザーである私だ**」ということである。つまりベストの解決方法はクリックのあるモードも付けるということなのである。それが**便利**ということなのである。

そもそもワープロメーカーは，クリック音がなんのためにあるかを考えたことがあるのだろうか。そして私のように，あれがないとついつい打鍵に余計な力が入ってしまい，だんだん指先に疲労がたまっていく人間もいることを知っているのだろうか。

繰り返す。クリック音が必要かどうかを決めるのはメーカーではない。ユーザーなのである。

一般にこの手のことはあちこちのワープロにいえることなのであるが，

親切設計

という点について，実に貧弱なのである。ここで，また一太郎を例に挙げるのであるが，一太郎の欠点としていつも指摘されるのがツリー構造をしたコマンド群である。たとえば検索は，［ESC］［S］［S］となっている。すなわち３回キーを押さなければならないのである。

ここからちょいと微妙な（そして本質的な）問題に入るのであるが，この一太郎の検索コマンドは明らかなタコである。しかし一太郎の設計者はタコだとは思ってないであろう。なぜならば根本的な部分で認識に差があるからなのである。すなわち，

1)　すぐ手の届くコマンド群か？
2)　整理されたコマンド群か？

の違いである。おそらくこれは思想の違いにまで達する問題なのである。

例を挙げよう。ものすごくごちゃごちゃした部屋に住んでいる人がいるとする。他人にとってはその部屋は混沌そのものであって，ほとんどゴミタメとしか見えない。しかしその部屋の住人はどこになにがあるのかを完全に把握しており，ゴミの山のなかから自由自在にものを取り出したりするのである。

その逆にキチンと整理されているにもかかわらず，ものの出し入れが面倒な部屋というものもあるだろう。すべてのものはタンスや机の引き出しや整理棚などにきちっと入っている。そしてその部屋の主は机のところに行き，上から３番目の引き出しを

開け，右端のトレイからハサミを取り出すのである。

両者の違いは「部屋は散らかっていてもいいから，とにかくものに素早く手が届くほうがよい」と考えるか，それとも「とにかく部屋は整理されていなければならない」と考えるかの差である。

はっきりいってどちらがよいかを証明することは不可能である。しかし，「たんすの上から５番目の引き出しの右奥」は，宝石箱を置いとくにはよいだろうが，テレビのリモコンの置き場所には不便であろう。もっとも私のように２日に一度は捜し回るようなことは避けられるだろうが。

英文ワープロからなにも学ばない

ここで英文エディタについて書くことにする。

前にも書いたことがあるが，私が愛用しているのはMINCEというエディタである。これはEMACS系に属するエディタである。その点からすると「名家の血を引く」といっても過言ではないだろう。実際「成り上がりのJAPの作ったワープロ」よりはずっと知的でエレガントである。

さて，ここまで読んでむっとした人も多いであろう。もっともである。しかしMINCEのほうが**本当に知的でエレガント**なのである。これは動かし難い事実なのだ。なにが知的であるか/なにがエレガントであるか，を証明することは不可能である。裁判で争うことも不可能である。ましてや多数決などは無意味である。しかし，この世には**知的なものが存在するし，またエレガントなものも存在するのである。**

「知的だとかエレガントだとかを，お前の

勝手で決めるな」という人もいるだろう。もっともである。だけど本質的にこの手のものごとは**主観だけ**がモノサシなのである。だから「勝手に決める」しかないのである。むしろ「勝手に決めるな」などというほうが無理難題なのである。

これ以上続けると泥沼にハマるよーな気がするのでやめておくが，ま，MINCEと日本語ワープロの差は，マドンナと本田美奈子ぐらいの差があるのだ。わかったか。

てなところで，MINCEと，いま私が使っているワープロ（ま，よーするに即戦力だ）のエディタとしての機能の差をちょいと見てみるわけである。以後，MINCEはM，即戦力はSとする。

語検索

S：［F1］［F1］［F1］［F6］+「捜す言葉」

M：CTRL-S +「捜す言葉」+［ESC］

逆向き語検索（カーソルのある行から文頭に向かって検索する）

S：なし

M：CTRL-R +「捜す言葉」+［ESC］

＊大事なのは即戦力では「捜す言葉」をいちいち入れ直さなければならないが，MINCEでは［ESC］だけだと直前に使った「捜す言葉」を思い出してくれることである。

画面の書き直し

S：なし

M：CTRL-Lで「カーソルのある行が画面の中心にくるように書き直す」

＊エディット行のカーソルの位置は変えずにカーソルの前後の行を見えるようにしたいときに極楽。

改ページ

S：ROLL UP/ROLL DOWN

M：CTRL-V/ESC+V

＊不思議なことではあるが，実はこんな単純な機能でもエレガントかどうかの差が出てくるのである。すなわち即戦力は完全に次の画面にいってしまうのであるが，MINCEは2行分残すのである。つまり私の使っているMINCEは1画面22行表示なのであるが，CTRL-V/ESC+Vによってスクロールするのは20行なのである。さて，この機能によるメリットはというと，これを言葉で表現するのはちょっと面倒なのであるが，MINCEのほうが文章の連続性を直感できるということがある。それに対して即戦力のほうは「パシッ」と切り換えられるので，どうもイラッとする面がある（ひょっとしたら私がMINCEに染まっているための偏見かもしれない）。

移動

S：［F1］［F1］［F2］+範囲指定+移動先指定

M：カーソルを複写元に持っていってCTRL-@（マーク指定），CTRL-Wでバッファに取り込む（削除される），移動先でCTRL-Y

複写

S：［F1］［F1］［F3］+範囲指定+複写先指定

M：カーソルを複写元に持っていってCTRL-@（マーク指定），［ESC］+［W］でバッファにコピー，複写先でCTRL-Y

＊文章で書いた分にはよくわからないであろうが，この移動と複写には両者の本質的な思想の差が明確に表れているのである。すなわち，即戦力では「移動」と「複写」はそれぞれ別個に独立した機能として用意されている。それに対してMINCEには「移動」,「複写」自体はコマンドとして独立しておらず，ほかのもっとシンプルなコマンドを組み合わせることによって実現するようになっている。すなわち，

CTRL-@：カーソル位置にマークを付ける（主な用途は範囲の指定であるが，そればかりとは限らない）

CTRL-W：CTRL-@で指定された位置から現在のカーソル位置までのテキストを削除し，それをバッファに入れる。

ESC+W：CTRL-@で指定された位置から現在のカーソル位置までのテキストをバッファにコピーする。削除はしない。

CTRL-Y：バッファの内容をカーソル位置に吐き出す。

ここで特筆しておきたいのは，実際はMINCEではESC+Wの動作を覚える必要がないということである（もちろん覚えておいたほうが便利であるが）。それはどういうことかというと，一度バッファに取り込まれた内容は，別の内容が取り込まれるまでは消えないので，CTRL-Wで削除したあと，すぐさまその場でCTRL-Yで復旧し，その後複写先までカーソルを持っていき，もう一度CTRL-Yを実行すれば「複写」になるのである。このことはそんなに便利な気がしないかもしれないが，私はCTRL-Wではなく，CTRL-K（1行削除）を使っての行単位でのコピーで多用して便利している（もっともアセンブラでのことだが）。

さて，これはひとつの仮説なのであるが，どうも日本人の「規則」に対する考え方はかなり歪んでるのではないだろうか。つまり，日本人は「規則を守ること」に関しては極めて情熱的なのであるが，どっこい「よい規則を作る」ということに関しては，とんでもないタコスケなのではないだろうか。そう考えるといろいろ思い当たるフシがあったりする。たとえば「いじめ問題」だけど，転校させることで解決しようとしても「規則で許されていないから」というだけで一蹴したりしているよーである。規則は人間のためのものなのであって，人間が規則のためにあるのではないのだ。しかしお役人様たちは別の見解のようなのである。

表1　いわゆるMINCEのコマンドサマリー

C-@	Set Mark	M-C-W	Make Previous Delete
C-A	Beginning Of Line	M-<	Beginning Buffer
C-B	Backward Character	M->	End Of Buffer
C-D	Delete Character Forward	M-A	Backward Sentence
C-E	End Of Line	M-B	Backward Word
C-F	Forward Character	M-C	Capitalize Word
C-G	Abort/Cancel Prefix	M-D	Delete Word Forward
C-H	same as <DEL>	M-E	Forward Sentence
C-I	same as <TAB>	M-F	Forward Word
C-J	Newline Insert, Indent same	M-H	Mark Whole Paragraph
C-K	Kill Line	M-K	Kill Sentence Forward
C-L	Redisplay Screen	M-L	Lowercase Word
C-M	same as <CR>	M-Q	Fill Paragraph
C-N	Next Line	M-R	Replace String
C-O	Open Line	M-S	Center Line
C-P	Previous Line	M-T	Transpose Words
C-Q	Quote Nest Character	M-U	Uppercase Word
＊C=Control Keys		＊M=Escape Keys	

つい脱線してしまったが，ようするにわれわれは規則を作るのが苦手な民族なのである。だからワープロのコマンドを決めるときには人一倍（民族一倍？）慎重になるべきなのだ。

もしもそれができない場合には，簡単な解決法もあるのである。すなわち「コマンドコンパチ」という手である。「なんだ，マネか」などといってはいけない。これは正しいソフトウェア設計法なのである。同じ操作でよりよいソフトなのである。

具体的にはWordStarなどのメジャーなエディタのコマンド体系をそのまま持ってきてしまえばよい。著作権上も問題ないので，本来なら大いになされてしかるべきなのだが，残念ながら日本のワープロソフトメーカーは「マヌケな独自性」に固執してしまっているようである。

高速エディタ

最近はそうでもなくなったが，ひと頃は「高速」を売り言葉にしたエディタが賑わっていた。確かにエディタが速いということはよいことである。しかし，これまた当たり前のことであるが**速いだけでもどうしようもない**のである。つまり，速さというのは**操作性の一要素にすぎない**のである。素晴しく速いが操作性が悪いエディタというものも存在しうるのである。

確かによい車の条件として，速いということがある。しかし速けりゃいいというわけではない。たとえばウイリアムズ・ホンダ謹製の車はものすごく速いそうであるが，速度以外のありとあらゆる要素はかなり劣悪であろう。なにせスピードメーターさえもついていないのである。ま，これはただ速けりゃいいってものではないという例である。

で，ワープロなのであるが，大事なのは総合力なのである。単一の機能が突出していてもだめなのである。その点はよろしく。

こまごま

ワープロメーカーの知的水準を窺い知るのに，ちょうどよいと思われるのが「ワープロのパッケージの大きさ」である。同じことを感じている人も多いのではないかと思うのだが，ここ最近は，ワープロソフトの入っている箱がどんどん巨大化しているのである。

確かに見やすくするためマニュアルが大判化しているということもあるのだが，しかし巨大化の主な要因はスポンジなどによる「上げ底」なのである。これは開けてみれば誰でもわかる。

さて，なぜこのよーなことになっているのかは「生産者心理」から明らかであろう。すなわちソフトウェア売り場で目立ちたいからである。また，やはり「生産者心理」としては，「客は大きいほうを好むだろう」というヨミがあるのだろう。

愚劣である。

外側だけを飾ったって無意味である。パッケージを膨らましたって邪魔になるだけである。どーしてわからないのだろうか。そんなことに使う頭と金があるなら，ほかにもっと行うべきことがあるのじゃないですか？　えっ？

マニュアルのでき

それからワープロメーカーにいっておきたいことのひとつに，マニュアルのできの悪さがある。伝え聞くところによると，日本民族というのは世界に例を見ないほど「マニュアルを作るのが下手」だそうである。

確かに交通標識とか，駅や空港の構内の道案内とかは恐ろしくわかりにくくなっている。もちろん最低限の機能は果たしているのだが，本当に最低限なのである。これはいったいどういうことなのかというと，ま，無責任に想像するに，鎖国の間はずっと異民族と接することがなかったために「いい加減な説明でも用が足りてしまう」状態に長く置かれてしまい，「説明することのノウハウ」が廃れてしまったからではないかと思われる。

また，東洋には無意識のうちにサービス業を低く見る傾向があるらしいので，それと関連して「親切に説明する」ということを軽んじるのかもしれない（本当は，よい説明とはかなり高い知性を必要とすることなのであるが）。直接的には国語教育で「話し方」や「実用文書の書き方」がまったく無視されていることがあるだろう（アメリカの大学ではそのための講座もあるそうな）。

ま，とにかく日本人の書くマニュアルがかなりひどいらしいのだけは事実である。その証拠としては，多くのワープロのマニュアルが結構ぶ厚いということが指摘できる。本当は，厚いマニュアルというのは，とても恥ずかしいことなのである。また書店には数多くの一太郎の解説書が並んでいるが，それも恥ずかしいことなのである。つまり標準で付属しているマニュアルがわかりづらいということなのだから（実は別の意味もあるらしいのだがそれはひとまず置いておく）。

結論

本当はワープロ（エディタ）を作るということはものすごく難しいことなのである。ただし難しいのはプログラミングではなく「設計」のほうである。どうすればもっと操作性がよくなるか/マニュアルを見る回数が少なくてすむか/すみやかに操作を覚えられるか/打鍵数を少なく抑えられるか/直感を逆なでしないか/操作に習熟したあとにうっとうしさを感じないか，などは非常に複雑で微妙な問題なのである。そして忘れてはならないのが，これらすべてのことに対してバランスを取らなければならないということである。たとえば，いちいちコマンドのメニューを表示するようにすればマニュアルを見る回数が少なくてすむだろう。しかしそれでは速度に影響するし，また，習熟したあとではうっとうしく感じられる。もしも打鍵数を少なくすることに徹すると，逆にコマンド体系の設計が難しくなっていくだろう。すみやかに操作を覚えられることを第一にしたはよいが，バランスを欠いたならばコマンド体系が単純になりすぎて，使い込むうちにイライラするものができるかもしれない。

そう，大事なのは，多機能ではなく高機能なのである。高速性ではなく操作性なのである。変換機能ではなく総合機能なのである。パッケージの大きさではなく中身の濃さなのである。累積販売数ではなく満足しているユーザー数なのである。

さてさて，あなたの使っているワープロは果たしてエレガントですか？

日本語

特集 迷宮の日本語処理環境

日本語のしがらみを越えて

ワープロは創造的活動のためのツールとしての価値を問われる時期にさしかかっている。しかし，ワープロを進化させられるかどうかは，ユーザーの道具に対する取り組み方にゆだねられているのである。

Saito Susumu
斎藤 晋

日本語ワードプロセッサについてどうこうというと必ず出てくるのが，「結局いまのワープロって清書マシンでしょ」とか，「これからは思考を助けるワープロが必要だ」とかいった話だ。もっとも，いまのワープロの場合は基本的なエディタとしての機能がどうしようもないので，思考を助けるどころか入力を助けることすらままならない状態ではある。つまり清書マシンとしても失格ということだ。まあ，そのあたりのことは前のページで祝氏が愛と感動の毒舌をふるってくれているようなのでこれ以上は触れないことにする。ここでは少しばかり前向きに（現状を省みずという意味で使われる便利な言葉だ）話を進めてみよう。

日本語ワープロは清書マシン?

結論からいえば，日本中にあるほとんどすべてのワープロは本質的に清書マシンである。もちろん，清書マシンといういいかたはよくないかもしれない。たいていの人はワープロに向かっていきなり書き始め，紙に下書きしたりしないだろう。私だってそうしている。ただ，明らかにいまの日本語ワープロは，「なんらかの理由で表現しなくてはならない文章が存在する」ということを仮定して作られているのである。これについてはワープロがツールとして利用される状況を考えてもらえればよい。

一般にワープロが使われるのは，社用の便箋に打ち出される案内状であったり企画書であったりすることが多い。パーソナルユーザーの場合は人それぞれだとは思うが，ま，多いのは葉書に印刷する季節の挨拶状だったりするのだろう。いずれにしても，これらはみんな人に見せるためにきちっとレイアウトして印刷されたものが目的の品となる。作家が小説を書いたり私がOh！MZの原稿を書く場合は普通の原稿用紙に書いて出すのと本質的に同じであり，どちらかというと例外である。それでも，編集者という人に見せる完成した文書を作るということには変わりない。ワープロ専用機のなかには原稿用紙のマス目に合わせて印刷できる機種が多い。

ここに示した例は，本来ワープロなんかなくったってもともと書かなくてはならないものばかりである。ワープロはそれらを効率よく綺麗に表現することができる。今のワープロはそのための機能が豊富にある。けれども表現する（清書する）以前に，その文章はユーザーが思考して作るのだという観点が抜けているとしか思えない。

「そんなことはない，私はワープロを思考ツールとして活用している」という人もいるだろう。が，それはあなたがとても偉いのであってワープロのほうにはあなたの思考を助けるつもりはない。そういう人はワープロがなくても手元のメモ用紙を思考ツールとして活用することだろう。

要するにここで本質的に清書マシンと呼ぶのは，ユーザーが考える文章を表現する機能はあるが，なにかを書きたいという衝動から，構想をまとめ，いくつかの表現をシミュレートしながら推敲を重ね，完成に至るといった過程がほとんど考慮されていないということだ。ここにきてアウトラインプロセッサと呼ばれるものが話題となり，これからは思考を助けるツールが必要という声が聞かれるのも当然だろう。

さらにグラフィックイメージによる視覚的なサポートも重要となる。

誰でもなにかを書こうとするとき，頭のなかにはなんらかのイメージがあるはずだ。人はこのイメージを言葉の集合に置き換えて紙の上に表現する。あるいは，すでに言葉によってイメージを形成している場合も多い。人間の考えは常に言語によって支配されているからだ。ワープロのありがたいところは，イメージの断片から書き始めてもあとでいくらでも編集可能であることだろう。しかし，人間の思考は視覚によるイメージによっても大きく影響を受けるということも問題だ。紙の上で下書きなどをする場合，キーワードをマルで囲ってみたり，アンダーラインを引いてみたりする。私の場合，アウトラインを考える際にはポイントとなる単語を手書きでレタリングしたりしてイメージを刺激するよう心掛けていたものだ（X68000のワープロならマウスを使っていろいろできるが）。

日本語処理の背景

で，清書マシンである。ただし，清書マシンであるからといってワープロを責めるつもりはない。問題は，人々がそれを求めたのであり，ワープロは求められて清書マシンになったということである（そんなもの求めた覚えはない，といってもダメ。あなたが求めていなくても，人々がちゃんと求めているのだから）。まずこのことをきちんと理解する必要がある。ワープロはもっともっと進化するはずだ。しかし，日本語ワードプロセッサというものが発達してきた背景を無視しては進化を語ることなどできはしない。私は現状は省みないが，過去はめめしく省みる。

日本語処理がこれほどまでに進化してきたのはOA化の波に乗ってのことだ。これに個人向けのポータブルワープロのヒットが拍車をかける。価格はどんどん下がり，機能は飛躍的に向上している。パソコンも健闘はしているが（ほとんど一太郎という気もする），ワープロ専用機はパソコンを駆逐する勢いで市場を伸ばしている。

これはじつに恐るべきことだ。技術者だってSF作家だってこんな事態は予測しえなかった。コンピュータに関しては，日本は常にアメリカのあとを追うように発展してきた。ところが，私の知っているかぎり，かつてアメリカで日本語ワープロが大流行したという話はない。それどころか，「WordStarに表計算機能とイメージ取り込み機能と豊富なフォントと強力な印刷機能とついでにアウトラインプロセッサをくっつけ

た専用電子タイプライタSuperSTARがIBMを追い抜き2千万台突破！」なんて話も聞かない。やはり日本語ワープロが売れるのは日本語のもつ特殊性によるものなのだろうか。

ここでちょっと実際に日本語ワープロが必要とされる状況を考えてみよう。

そもそも日本語で文章を書くというのはじつに難しく高度な技術を要する。いうまでもないことだが，日本語の文章には，漢字，ひらがな，カタカナ，数字，そして場合によっては英字までも駆使しなくてはならない。コンピュータにとって難しいのは当然，人間様にとっても大変なことなのだ。だが，私たちはこの日本語で文章を書くということから決して逃れられない。学生には常にレポートがつきまとい，会社に入れば挨拶状[1]だの報告書だの始末書だのとうんざりするようなものが待ち受けている。

そして，なんといっても最大のドラゴンは漢字である。漢字が使えることによって得られるメリットは計り知れないものがあるが，正しく使うのはやはり難しい。だいいち漢字は覚えることも大変だが，使っていないと簡単に忘れてしまう。そして1画1画手で書き込んでいくのはとてつもなく大変な作業である。タイプライタを使用する欧米人が，26文字のアルファベットを10本の指を使って機関銃のように打つことに比べれば，圧倒的に不利である。そこへもってきて多くの人は決して字を書くのが上手ではない。ここだけの話だが，私は編集室の人から清水和人氏の生原稿を見せられ，これなんて読むんだろうと相談を受けたことが何度かある。

さらに困ったことに，字を見ればその人の人柄がわかるなどと大真面目にいう奴がいることだ。こんな迷惑な話はない。かつて私は血のにじむような努力の末，変体少女文字を体得したが，それで私の人柄が変わったなどという人はいない。きっと多くの哀れな社会人はワープロの普及によって自分の字が人目に触れなくてもよい時代がくることを待ち望んでいたに違いない。

というわけでだいたい様子がわかってきたことと思う。私たちはコンピュータにさまざまな新しい可能性を求めている。ワープロの普及が新たな知的で創造的な活動の場を広げてくれるものと期待している。だが，商品としてのワープロが普及してきたのは，決してそのような期待に応えてのことではない。日本語ワードプロセッサという商品は，世の中のしがらみから私たちを解放するために生まれてきたものだったのだ。かな漢字変換[2]は日本人の知恵，必要は発明の大母神ガイアである。

データとエディタ

さて，3割4割引きは当たり前で，デパート，電器街，大型カメラ店などに繁殖するワープロ専用機だが，メーカーによる操作環境の違い，データの互換性のなさ，プリンタの規格の違いなどはいっこうに改善される気配もない。キー操作の違いはパソコンのワープロソフトでもめちゃくちゃだが，データはMS-DOSフォーマットが主流となり，プリンタはセントロニクスというわけで，専用機に比べてはるかに救いようがある。

専用機がこのようなデタラメなかたちで普及してしまったのは，明らかにキーボード文化に対する認識の低さ，データに対する認識のなさによるものだ。

そして私は次の結論を持っている。エディタの重要性が生きてこないのは，文書に対するデータの概念が欠落しているからであろう，ということだ。コンピュータはソフトがなければただの箱だが，ソフトウェアにデータがなければ，たとえを思いつかないほど無意味なものとなってしまう。ソフトウェアを規定するのがデータの概念といってもいい。このことはMacintoshの環境を追ってみるとよくわかる。すなわち，

文章はデータである
言葉もデータである
紙1枚もデータである
そして，思考もデータである

ということだ。「そんな，言葉は生き物でしょ」なんていってくるのはそもそもデータの重要性がわかっていない証拠である。データは蓄積され，分類され，混合され，反応し，還元し，そして新たなデータを生み出すのだ。生き物だって，化け物だってかまわない。コミュニケーションが大切なのも，外人さんが討論が好きなのも，データのぶつかり合いによって，より進化したデータの集合を得るためである。

1) しかもあの「貴社ますますご清栄のこと……」とか「平素は格別のご高配を賜り……」で始まる文面である。すでにこのテの定型文書はほとんどの会社でワープロ化されているので，出すほうは宛名と用件を書き込めばいいだけだし，受け取ったほうでも挨拶の部分はパターン認識ですむ。しかしちょっと冷静に読めば，これらはお決まりのパターンであるにもかかわらず，自分の世界の言葉とは思えない不自然さをもっていることがわかるだろう。試しに手書きで書いてみるといい。自分がものすごく無能な社員に思えてくる。日本語ワープロが普及した訳がなんとなくわかるはずだ。

2) かつて，一太郎が連文節変換の日本語フロントプロセッサATOK 4を搭載して登場してきたとき，その一太郎をさかんに私にすすめる男がいた。「朝日新聞の記事ならほとんど一発で変換できるんですよね」。これである。こういう場合に使われるのは必ず新聞，それもなぜか朝日でなくてはならないらしい。いったい誰が新聞の記事なんか入力するんだろうと思いながらも「ホゥ」ってなもんで聞いていた。知恵の結晶とはそのようなものなのか。

文章はデータである。だから自由にエディットできなければならない。言葉もデータである。だから，自由に変換できなくてはならない。紙1枚もデータである。だから，ビットイメージで処理されなければならない。そして思考もデータである。だからこそ，自由に構成できなければならないのである。

どうすればよいのか

いったい私たちはどうしたらよいのか，X1turboのShogunはいつになったら発売されるのか，私はその筋キーホルダーをもらえないのか。

まだ私たちには限りない可能性が残されている。ワープロは既成の概念，既成の生活，既成のビジネス，そしてコンピュータに対する乏しい認識によって支配されてきた。しかしこれからは違う。ひとりでも多くのパーソナルユーザーが日本語処理環境の可能性を考え，そこに新たな価値観，新たな認識を養っていけばよいのである。われわれにとって必要なことは，なにもワープロの新しい用途を考えることではない。大切なことは，データに対する認識である。結局，文章を書くということの質的な変化が起こらなくてはならないのではないか，文章を書くという人間の態度に質的な変化が求められているのではないか，ということなのだ。オリンポスは滅び，これからはみんな自分で生きていくのである。

日本語

特集 迷宮の日本語処理環境

X68000ワープロ/エディタ

Wordprocessor Report

X68000が用意してくれた日本語処理環境は，果たして私たちユーザーを満足させてくれるものなのだろうか。まだまだ未知数のX68000と日本語処理，その2つの関係について探ってみたい。

Nakagawa Norichika
中川 智哉

いつでもどこでも日本語入力

いま，私はX68000の日本語ワードプロセッサの上にいます。マウスカーソルが縦横に走り回る，なかなか近代的な風景です。今回はここから，X68000に標準で用意された日本語ワードプロセッシング環境についてリポートしてみたいと思います。

まず第一のポイントは日本語フロントプロセッサ（以下，日本語FPと略す）。文字入力ができるところではどこででも，これを使って日本語入力が可能です。ワープロはもちろん，エディタ，BASIC，ビジュアルシェルのNote Bookや電話帳。辞書が共通に使えること，操作法が統一されることは，想像以上にありがたいものです。

とはいっても，じつはワープロだけは少しばかり操作法・有効キーが異なるのです。それを表1にまとめてみました。ふだんなにげなく使っていたのに，調べてみるとマニュアルに載っていなかった，というようなことも入れてありますので，リファレンスとして利用できると思います。

このように，標準の日本語FPとワープロで操作法が異なるのは残念なことです。特にワープロ時，「辞書登録」が名詞しか登録できない，「表示選択モード」が数字キーでしか選択できないなど，より高度な日本語処理が期待されるワープロで基本的な機能が劣るのは本末転倒ではないでしょうか。

この日本語FPは連文節変換が可能で，かなをどんどん入力していけば変換キーを押さなくても自動的に漢字に置き換えてくれる「逐次自動変換」と，変換キーを押すことで文章をまとめて変換する「一括変換」の2種類の変換モードがあります。内容的には「一太郎」の日本語FP，ATOKなどとも十分張り合えるだけのものをもっていますが，多少気になる点もあります。

まず学習機能がやや弱い感じがすること。はっきりとした分析はできませんが，使っていて「あれ？」と思うことがあります。それからスピード。キー反応，変換ともにです。特に，ときどき10数秒間黙りこくってしまうことがあるのは，ガーベジコレクションかなにかをやっているのでしょうか。辞書をRAMディスクに転送して使ってもそこそこしか速くならないことから考えると，まだ改善の余地があるでしょう。

あと，できればカタカナや半角文字，英字（ローマ字）で入力しても変換してくれるとありがたいですね。また，細かいことかもしれませんが，実際に文章を書いていると助詞や活用語尾などから打ち始めてしまうことがあります。そんなとき「無変換」（1字無変換）がサポートされていたらなあ，と思うのです。

日本語FPは基本的なユーザーインタフェイスのひとつなわけですから，さまざまな人間からさまざまな要求が生まれてきます。いずれこのFPにも上位バージョンが登場してくるでしょうが，そういった要求をできるかぎり汲み取って，ユーザーに優しいFPになってほしいと思います。

スクリーンエディタの機能

今度はスクリーンエディタED.Xにやって来ました。ちょっとWordMasterに似ていますが，こちらのほうがはるかに高速・

■表1 かな漢字変換のキー操作(この表はすべてワープロで作成しました)

	機能	日本語フロントプロセッサ	日本語ワードプロセッサ
読み入力モード	漢字変換	XF3	XF3（逐次） XF1〜3，SHIFT-XF1〜3（一括） XF1〜5，SHIFT-XF1〜5（無・再変換）
	ひらがな/カタカナ変換	XF4	XF4（一括）
	全角/半角変換	SHIFT-XF4	SHIFT-XF4（一括）
	全文確定	XF5，ENTER	XF5，ENTER
	カーソル移動（左/右）	←/→	←/→
	インサートモードON/OFF	INS	（常にインサートモード）
	バックスペース/デリート	BS/DEL	BS/DEL
	変換エリア内カーソルホーム	HOME	
	変換エリア内クリア	CLR	
	入力キャンセル		ESC
漢字変換モード	次候補→表示選択モード	XF3，SPACE	XF3，SPACE
	表示選択モード		↓
	ひらがな/カタカナ変換	XF4	XF4
	全角/半角変換	SHIFT-XF4	SHIFT-XF4
	全文確定	XF5，ENTER	XF5，ENTER
	前文節移動	XF1	XF1
	次文節移動	XF2	XF2
	文節短縮	SHIFT-XF1	SHIFT-XF1，←
	文節伸長	SHIFT-XF2	SHIFT-XF2，→
	変換結果キャンセル	ESC	ESC
表示選択モード	次候補	XF3，SPACE，→	
	前候補	←	
	次候補群	↓	↓
	前候補群	↑	↑
	反転候補選択	XF5，ENTER	XF5，ENTER
	数字候補選択	数字キー	数字キー
	前文節移動（→漢字変換）	XF1	XF1
	次文節移動（→漢字変換）	XF2	XF2
	文節短縮（→漢字変換）	SHIFT-XF1	SHIFT-XF1
	文節伸長（→漢字変換）	SHIFT-XF2	SHIFT-XF2
	表示選択キャンセル	ESC	ESC
特殊	文字選択	記号入力キー	記号入力キー，メニュー
	辞書登録	登録キー（全品詞）	登録キー，メニュー（名詞のみ）
	コード入力モード体系切換	F7	
	辞書ドライブ設定	F8	
	ディスク/メモリ学習切換	F9	メニュー
	変換モード選択	F10（逐次/一括）	メニュー（逐次/一括/無変換）

読み入力モード：変換エリアに文字を入力している状態
漢字変換モード：漢字変換が行われ，注目文節が反転表示されている状態
表示選択モード：変換候補がずらずらっと表示されている状態

高機能です。スクロールだってビュンビュンですし，キー入力がバッファにたまってしまうなんてこともありません。もともとはアセンブラなどのテキスト入力のためのものですが，もちろん日本語FPが利用可能で文章の入力にも使えます。

このエディタについて取り上げるのは今回が初めてだと思いますので，主な特徴を個条書きにしておきましょう。

- 編集画面は横96または64文字，縦30行
- 横スクロールにより，画面端で折り返されることなく長い行が入力可能。1行の文字数(半角)は128,256,512,1024から選択
- 同時に10個までのファイルを編集可能
- カットバッファ（作業用文字列格納エリア）を介し，テキスト移動/コピー，ファイル間転送が可能
- 子プロセスが起動できる
- キーボードマクロが設定・実行できる
- CTRL/ESC/ファンクションキーによる多彩なコントロール

ここでいう“1行”とは必ず末尾に改行コードが入っています。設定された文字数いっぱいになったら強制的にCR+LFを入れて改行するので，行と行がつながっているということは存在しえません。

子プロセスの起動とはX-BASICにもある機能で，たとえばエディタからディレクトリをとったり，ファイルを削除したりなど，COMMAND.Xから実行できることはすべてできるといっていいでしょう。COMMAND.Xやビジュアルシェルを起動することもでき，EXITでちゃんとエディタに戻ってきます。メモリが足りればグラディウスやワープロだって実行できるのです。

キーボードマクロに関してはマニュアルにもまったく解説されておらず，エディタのヘルプメニューで出てくるだけですから，ちょっと詳しく説明しておきましょう。

このエディタでは一連のキー操作を覚えさせて，UNDOキーを押すだけで同じ処理を実行することができます。具体的には，キー操作を開始するところでESCと@を続けて押します。「マクロ定義中」という表示が出ますから，覚えさせたい“一連のキー操作”をします。最後にもう一度ESCと@を続けて押せば定義終了です。あとはUNDOキーを押すたびに“一連のキー操作”を行うのと同じ結果が得られます。

このエディタを十分に使いこなすにはそれなりの習熟が必要です。ユーザーのなかには，HELPキーを押すとドドッと出てくるヘルプメニューを見てビビッてしまい，それ以来エディタは立ち上げてもいない，という方もいるのではないでしょうか。その点マニュアルのほうも初心者には不親切な感じがしますね。

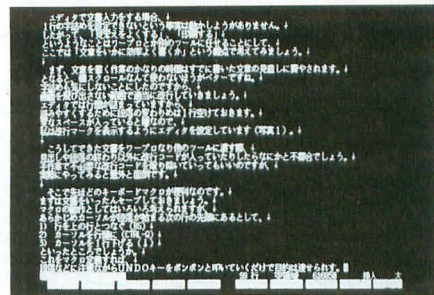
しかし，このエディタをテキストエディタとしてだけしか使わない，ひょっとすると全然使わない，というのはじつに惜しい。なにしろ，入力だけならたぶんワープロより速い（習熟すれば間違いなく速い），検索や置換が強力，辞書登録がすべての品詞で可能，必要なら子プロセスでDICM.Xを起動して辞書整備ができる，標準メモリでも辞書ディスクをRAMディスクに転送して使える（ワープロはビジュアルシェルを使わないようにしないとできない）などなど，おいしい部分がいっぱいあるのですから。

また，使い方が難しそうとはいっても，起動方法と終了方法(ESC+Q)，それにファイル入力（ESC+Y），ファイル出力（ESC+H）さえ知っていればなんとか動かすことはできますから，あとは使いながら覚えていけばいいのです。

文章のベタ書き用ツールとして

エディタで文書入力をする場合，1行の字詰めを設定できないという事実は動かしようがありません。したがって，「見栄えをよくする」，「印刷する」というようなことはワープロとか別のツールに任せることにして，ここでは「文章をいかに効率よく書くか」という観点で考えてみましょう。

まず，文章を書く作業のかなりの時間はすでに書いた文章の見直しに費やされます。となると，横スクロールなんて使わないほうがベターですね。字詰めも気にしないことにしたのですから，画面を飛び出さない範囲で適当に改行していきましょう。エディタでは行間が詰まっていますから，読みやすくするために段落の変わりめは1行空けておきます。変なスペースが入っていると嫌なので，私は改行マークを表示するようにエディタを設定しています。

こうしてできた文書をワープロなり他のツールに渡す際，見出しや段落の終わり以

外に改行コードが入っていたりしたらなにかと不都合でしょう。手作業で不必要な改行コードを取り除いていってもいいのですが，実際にやってみると意外と面倒です。

そこで先ほどのキーボードマクロが便利なのです。まずは文書をいったんセーブしておきます。マクロの動作は，カーソルが段落の始まる次の行の先頭にあるとして，

1) 行を上の行とつなぐ（BS）
2) カーソルを行頭に（CTRL-Q）
3) カーソルを1行下げる（↓）

といったところでしょうか。試しにやってみるとうまくいきました。これをマクロ定義すれば，段落などに注意しながらUNDOキーをポンポンと叩いていくだけで目的は達せられます。

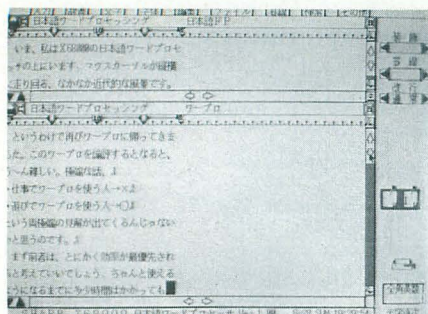
ここで注意しなければならないのは，ひとつの段落を1行に収めてしまうわけですから，“1行”の最大文字数が少なすぎると段落が収まりきらない，ということです。エディタ起動時に「-M1024」オプションを指定しておくのがよいでしょう。これなら1段落が全角文字で500字，本原稿（19字詰め）でいうなら26行までは大丈夫ですね。

キーボードマクロは文章を最終的な字詰めに適当に揃えて，だいたいの行数を調べたりする場合にも活用できます。やり方は皆さんで考えてみてください。それから，「キーボードマクロなんて使わずにプログラムで自動的にやったほうがいい」という方もいるでしょう。確かにそのとおり。頑張ってください。

日本語ワードプロセッサ

というわけで再びワープロに帰ってきました。このワープロを論評するとなると，う～ん難しい。極端な話，

- 仕事でワープロを使う人→×
- 遊びでワープロを使う人→○

という両極端の見解が出てくるんじゃない

かと思うのです

まず前者は，とにかく効率が最優先されると考えていいでしょう。ちゃんと使えるようになるまでに多少時間はかかっても，最終的に高速に文書入力ができるようになるなら“よし”となります。当然，ブラインドタイプにも追従できるだけのレスポンスが必要です。コントロールキーはもちろん，場合によってはファンクションキーを活用すべきでしょう。

その点，日本語FPとの絡みもあるのでしょうが，このワープロのキー反応はいまいちです。コントロールキー/ファンクションキーも使えません。かなり基本的な機能でもマウスおよびプルダウンメニューで行うことになります。今回，ED.Xのほうも紹介したのは，このワープロの日本語エディタとしての貧弱さに起因するのです。

いま何ページ目の何行にカーソルがあるのか，常に表示されていないのも不便です。最下行のタイトルを取っぱらってそういった情報を表示すべきではなかったでしょうか。ただし，ばかばかしいと思われるかもしれませんが時刻表示はありがたい。ワープロを打っているとついつい時間を忘れてしまうんですよね。

スクロールはかなり高速です。マウスを使ったカーソル移動，スクロールバーによる長距離ジャンプも便利です。そして，なんといってもマウスによるカット/コピー&ペースト（移動/複写）は快適で，使っていて嬉しくなってきてしまいます。

このワープロでは「章」というものが用意されており，画面を上下に分割して2つの章を並行して編集が行えます。表示行数が少ないのでさほど便利だとは思いませんが，カーソルの長距離移動の変形と考えると積極的に使う価値はあるでしょう。各章には名前や書式が設定でき，「章編集」によって順番入れ替えや削除などが可能です。

セーブ/ロードは，装飾や罫線などをすべて含んだものは文書単位で，いわゆるテキストファイルは章単位で行われます。ちょっとしたアイデアとして，文章の一部を誤って削除してしまった場合など，新しく章を開いて，セーブしてある文書を「ファイル入力」で通常のテキストファイルとしてその章に読み込めば簡単に修復することができます。そう，文書ファイルはテキストファイルとしても読み出すことができるのです（多少ゴミが出ますが）。

プリンタへの出力は“遊び”にとっても非常に重要な部分です。ここではすべての文字を24ドットのビットイメージ印字で出力するので，JIS第2水準漢字ROMを持っていないプリンタでもちゃんと打ち出すことができます。また，コードが一致しないために画面に出ているのと違う文字が打ち出されてしまう，な～んてこともありません。

ただし，問題もいろいろあります。まず，文字間隔が密着/狭い/普通/広いの4通りしかないこと，改行幅が1/2～1/6インチの5種類しかないこと。これは確かに安直に使えますが，パーソナルユースのときこそドット単位の細かい指定がしたいものです。あと，章単位にしかプリントアウトできないのは，章ごとに書式が変更できるからでしょうか。また，最近の多くのワープロに用意されている「袋とじ印刷」，「レイアウト表示」ができません。特に後者はX68000の本領を発揮できるところだけに残念です。

なお，ヘッダやフッタとして，文書名/章名/ページのいずれかをプリントできるのはいいのですが，その場合は本文の行数が書式設定の「一頁行数」より少なくなってしまうことに注意しましょう。また，マニュアルには載っていませんが，ファンフォールド紙（いわゆるプリンタ用紙）を使うときは「カットシートフィーダあり」にすればいいことは皆さんお気付きですね。

ワープロマニュアル読んでますか

さて，遊びでワープロする場合は効率よりも，操作が簡単で，仕上がりの美しいことがもっとも重要といえるでしょう。ここでX68000ユーザーの方に質問です。ワープロのマニュアルをちゃんと読みましたか？

失礼しました。じつは，私はこの原稿を書くまでほとんど読んでいなかったのです。日本語FPの使い方さえ知っていれば，あとはマニュアルレスでもなんとなく使えてしまう。それがこのワープロの最大の長所（イージーオペレーション）であり，最大の欠点（そこからあまり前進しない）ではないでしょうか。

ビットマップの特長を生かし，各種の字体（横倍/縦倍/4倍角/斜体/強調/回転およびそれらのコンビネーション），1/4角文字，ルビ文字，およびこれらに対する装飾（5種類の網かけ/6種類の下線/3種類の打ち消し線/文字囲い/枠囲いのいずれか），おまけに12種類の罫線が，画面16ドットとプリンタ24ドットの違いこそあれ，プリントアウトされるイメージどおりに画面表示される（縦倍/4倍角を除く）のは，これだけで何日も遊べるほど楽しいものです。

さらに，プリントアウト時の改行量は通常改行/改ページのほかに，上合わせ/最小改行/半分改行/重ね打ちを行単位に指定できます。しかも，それらがマウスでチョイチョイと指定できてしまうのです（ただし，1/4角とルビ入力の方法には改善の余地がありそうです）。

これだけの表現力を持ったワープロは少ないと思われますし，それらがこんなに簡単な操作で使えるものはもっと少ないでしょう。その観点でいえば“さわって楽しい”ワープロということができるのです。

以上，“仕事/遊び”と強引に分けて見てきました。現実にはこんなふうにわりきれるものではないでしょう。しかし，メーカーが，標準で，一般ユーザーが“楽しめる”ワープロを用意したことは評価にあたいすると思います。ほとんどのユーザーにとって，“ビジネスワープロ”なんか必要ないのですから。

私見ではありますが，今後メーカーやソフトハウスに期待したいのは，

1）仕事にも使える本格的ワープロ
1'）強力な日本語エディタ
2）画像情報も扱えるイメージワープロ
2'）カラー対応のグラフィックワープロ

などなどです。当然，日本語FPのパワーアップも随時受け付けております。その際は，すでに鍛えた辞書をなるべく生かせるようにしてくださいね。

特集 迷宮の日本語処理環境

MZ-2861書院28

Wordprocessor Report

Gotou Takayuki
後藤 貴行

MZ-2861に標準装備された「書院28」。これまでワープロ専用機としては実力を発揮していた書院だが，その日本語処理機能をMZ-2861上でどこまで優位に展開できるか，ぜひ注目しておきたい。

ワープロ専用機として定評のある「書院」が，ワープロソフトウェア「書院28」としてパーソナルコンピュータ上で走り出した。システムは2HDディスク1枚には収まらない大きさであり，2枚に分けられて供給される。キーボードには，「書院28」に対応したいくつかの特殊キーが設けられている。ここでは，「書院28」版日本語入力フロントエンドプロセッサの使いごこちはどうか，MS-DOSとの融合性は良好か，編集機能および付加的な機能は充実しているか，などの点に注目しながら「書院28」を使ってみた。果たして「書院28」は「書院」を超えるか。

日本語入力

まずことわっておく必要があるだろうが，「書院28」では，先月号で紹介したMS-DOS上で直接作動するMZ-2861用日本語入力フロントエンドプロセッサとはまったく異なる方式によって日本語入力を行う。「書院28」そのものはMS-DOSから立ち上がるのであるが，デバイスドライバとして供給される日本語入力フロントエンドプロセッサにはまったく依存しない。

入力は，カタカナ，ひらがな，英数のいずれについても全角と半角で入力が行える。

■図1 日本語入力モードの違い

「書院28」

変換	キー
ひらがな→漢字	確定前(変換キー)
ひらがな⇄カタカナ	確定前(機能＋SF1,SF2)
英数→ひらがな	確定前(機能＋SF1)
英数→カタカナ	確定前(機能＋SF2)
半角⇄全角	確定後(機能＋半換,全換)

・上記以外の変換は不可能

「MS-DOS用フロントエンドプロセッサ」

任意のモードで入力可能	→全角/半角	(F4キー)
	→ひらがな	(F1キー)
	→カタカナ	(F2キー)
	→英数	(F3キー)
	→漢字	(変換キー)

このうち，変換キーを押して漢字に変換できるのは，ひらがなモードの全角と半角のみである。なお，MS-DOSの日本語入力フロントエンドプロセッサでは，カタカナであろうと英数であろうと，漢字に変換することができた。

カタカナや英数モードで入力した場合でも，いったんひらがなに変換したあとなら，変換キーを押して漢字に変換できる。ただし，英数→ひらがな変換(機能＋SF1)→漢字変換(変換)，というちょっと複雑な操作が必要である。一方，ひらがなモードで入力した文字を，英数文字に逆ローマ字変換することはできない。このため英数文字を入力するときは前もってモード切り換えを行うということになる。カタカナとひらがなの変換は，確定前なら自由に行えるが，半角と全角の変換は確定したあとでないと行えない。以上をまとめると図1のようになる。

この図から明らかなように，「書院28」の入力方式では，「入力前にモードを切り換える」必要がある場合がでてくるため，モード切り換えを忘れてキーを叩いてしまうと，入力結果が無駄になる可能性がある。ただし，英数文字をまったく含まない文章であれば，これらの心配は無用である。ひらがなからカタカナへの変換は自由に行えるのであるから，ひらがな入力モードで入力していけばよいからだ。

結局，「書院28」の日本語入力方式は，専用機「書院」との互換性を保つために採用されているらしい。BASICやリレーショナルデータベースなど，ワープロ以外のアプリケーションでは，日本語と英数文字を頻繁に切り換えて使用することが多い。そのたびにモード切り換えを行っていたのでは能率が悪い。このため，MS-DOSで使用される日本語入力フロントエンドプロセッサには，VJEやATOKライクなものが採用されたと思われる。

キーボードについても若干触れておく。専用機「書院」のキーボードは，普通のキーボードでスペースキーがある位置に変換キーと無変換キーが配置されていた。このため，両手をホームポジションに置いたまま，右手の親指で変換キーを，左手の親指で無変換キーを押すことができ，きわめて軽快に入力作業を行うことができた。英語ではスペースが単語の区切りとして重要な意味を持つが，日本語ではさほど重要でないため，スペースキーは小さくともよいのである。専用機「書院」の設計は，このように言語の本質までついているといってよい。なおMZ-2861のキーボードはスペースキーが少し大きめであり，ホームポジションのままで変換キーを押すには無理がある。

編集機能

「書院28」には，専用機「書院」でつちかわれてきたさまざまな編集機能がある。これらの機能は文字入力を行いながらダイレクトに利用できるため，必要な機能を見つけやすい。すなわち，多くのワープロソフトウェアで見られるような多重メニュー方式を採用せず，ファンクションキーや機能キーの1つひとつに，それぞれ異なった機能を割り当てている。このためユーザーは，望みの機能をただちに引き出して使用することができる。なお，機能キーそのものはキーボード上のコントロールキーに割り当てられているため，コントロールキーによるカーソル移動は行えない。まっとうなエディタに慣れたユーザーは，とまどうことがあるかもしれないが。

たとえば，ある単語に下線を引く場合には，機能＋“(”を押すだけでよい。さらに親切なことには，“(”キーの前面に「アンダーライン」と記入してある。

これに対しメニュー形式のワープロソフトウェアでは，まずエスケープキーなどでメニューを呼び出して，それから迷路のような多くのメニュー群の中を行ったり来たりしなければならない。下線ひとつ引くのにメニューを呼び出さなければならないのは，収入印紙を貼った申請書を持って焼き芋を買いに行く光景に似ている。

機能キーを用いて行われる編集機能は，下線や4倍角をはじめとして30種類以上ある。ファンクションキーにもいくつかの機能が割り当ててあるため，通常の操作は次のように行われると思えばよい。

機能キー＋一般のキー
機能キー＋ファンクションキー
シフト＋ファンクションキー
ファンクションキー

なお，ひとつのファンクションキーに最高4つの機能が割り当てられており，ファンクションキーのF6でメニューを切り換えて選択を行う。このため，必要な機能を探すのに若干手間取ることがあるかもしれない。

カーソルの移動は，かなりきめこまかく行うことができ，機能キーとカーソルキーを組み合わせて使用することにより，各行の右端左端，文頭文末などへ瞬時に移動できる。カーソルが文頭にあるときに，機能キー＋3（文末）を押して文末へ移動させると，文書の大きさが最大（約40Kバイト）の場合で約20秒かかる。英文ワープロWordStarと同程度の速さ（遅さ？）である。ただし文末から文頭へのジャンプは一瞬に行われる。

文書の大きさは最大で約40Kバイト（全角文字で2万字）であり，「一太郎」や「デスクup」などMS-DOS上で働くワープロソフトの中ではかなり小さめである。しかし，専用機「書院」のWD-530/630（最大16Kバイト）と比べれば若干大きい。

差し込み・切り貼り・分類

なんのことかわからないと思われるが，この3つこそ「書院28」のもっとも重要な機能である。まず差し込みとは，正確には差し込み印字のことであり，同じ手紙を多くの人に出す場合などに使用される。本文は同じ文面にしておき，相手の名前だけを変えて何枚も印刷できる。英文ワープロWordStarのオプションであるMailMergeと同じようなものである。ただしMailMergeでは，名前と住所，日付など複数の項目にわたって入れ換えを行うことができるのであるが，「書院28」の差し込み印字では，文書中の1カ所（手紙の宛先など）を入れ換えられるだけとなっている。

切り貼り機能は，編集中の文書の一部を切り取ってセーブしたり，あるいは他の文書ファイルをディスクから読み出して，編集中の文書に取り込むためのものである。編集中の文書内で切り貼りを行うこともでき，この場合はいったん領域指定を行ってから移動コマンドか複写コマンドを使用する。

分類機能は，住所録などのデータファイルから条件付き検索を行ったり，ソートしたりするためのものである。さらに，データファイル中の数値項目の合計を求めたりする機能も備えている。すなわち，「書院28」は単なるワープロソフトウェアではなく，データベース＋表集計の機能も若干兼ね備えた統合化ソフトウェアであるともいえるのではないか。

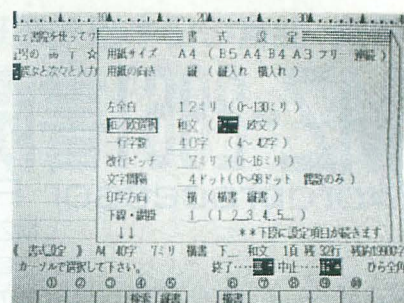
なお，専用機「書院」WD-630では，オプションのソフトウェアにより，図形やグラフの作成や表集計（カルク）などがサポートされていたが，「書院28」では残念ながら図形は扱えない。

システムの構成

「書院28」のシステムディスクは2枚に分けられており，1枚目のディスクには，「書院28」の実行可能ファイル“shoin28.exe”や，プリンタ設定用ファイルなどが入っている。驚くべきことにshoin28.exeの大きさは460Kバイトを超える。数式処理システムREDUCEのMS-DOS版実行可能ファイルと同程度であるというのだからおそれいる。そのうえ，2枚目のシステムディスクには，さらに400Kバイト近くのオーバーレイファイル（BASICでいえばCHAIN命令で読み込まれるファイルのこと）が入っており，その他のコマンドファイルも合わせるとシステム全体の大きさは1Mバイト程度となる。このため，「書院28」の起動には2枚のシステムディスクが必ず必要であり，リセットから初期画面の出るまでに30秒ほどかかる。

すると専用機「書院」では，この1Mバイト近いシステムがすべてROM化されて入っているというのであろうか。こちらにもあらためて驚いてしまう。なお，専用機「書院」ではシステムがROM化されているため立ち上がりはほぼ瞬時である。

「書院28」が立ち上がったあと，Aドライブのシステムディスクを文書ディスクと取り換えて，文書を読み込むことになる。ここで注意したいことは，「書院28」は専用機「書院」と同じようにあくまで内蔵ドライブA，Bでの使用を前提としており，拡張ドライブやRAMディスク，ハードディスクなどのサポートは一切なされていないということだ。もちろん，shoin28.exeをハードディスクに転送して立ち上げることは可能だが，ファイルコンバータなどを使用する際には必ずAドライブへ読みに行くので，結局ディスクをとっかえひっかえ使用することになる。

文書ディスクを作成するには，MS-DOSのFORMATコマンドでフォーマットしたあと，さらに「書院28」の文書ファイル初期化コマンドで初期化を行う必要がある。文書ファイル初期化コマンドは，ディスク中に，以下のようないくつかのファイルとディレクトリを作成する。

INDEX	512バイト
TEMP	51200バイト
PRG	51200バイト
NAME	〈ディレクトリ〉
BANGOU	〈ディレクトリ〉

文書ファイルは，これらのファイルを通して間接的に取り扱われるため，MS-DOSに戻ってDIRコマンドで中身を見ても，通し番号のついたファイル名がいくつか見えるだけである。ユーザーは階層化ディレクトリによるファイル管理を許されず，各文書ファイルにつけられる「通し番号」と「見出し」と「備考」のみを頼りに作業を行う。MS-DOSの階層化ディレクトリを理解しなくともよいので，初心者にとって最初のうちはわかりやすいかもしれない。また，会社などで多くの定型文書を用意しておき，必要な際に取り出して使用しているというようなケースでは，「○○番の文書を印刷して下さい」などと簡潔な指示が行えるため使い勝手がよいと思われる。

文書ファイルの中身は，シフトJISではなく，「書院28」独自のコード体系を使用している。このため，MS-DOSのTYPEコマンドで文書ファイルの中身を直接見ることはできない。アスキーコードおよびシフトJISのファイルを扱うには，次に述べるファイルコンバータを通して変換を行う必要がある。

ファイル変換

「書院28」は，付属のファイルコンバータによって，シフトJISコードの文書ファイルおよびワープロ専用機「書院」の文書ファイルを扱うことができる。これにより，BASICプログラムなどを，「書院28」で書くといったことも不可能ではない。

このファイルコンバータは「書院28」上で作動するため，文書編集中でもMS-DOSに戻ることなく変換作業を行える。変換の手順はきわめて簡単であり，「書院28」文書ファイル→シフトJISファイルの変換方法を以下に示しておく。

日本語

特集 迷宮の日本語処理環境

WD-260F/540/5010D

Wordprocessor Report

Satou Tomohiko
佐藤 友彦

今回ここでは，WD-260F/540/5010Dの３タイプをご紹介するが，細かい機能は省略してシャープのワープロ専用機が実務レベルでどこまで成長してきたかをポイントにレポートをお届けする。

大型チルト式ディスプレイが魅力 WD-260F

このWD-260F(138,000円)に触れてみての第一印象は，とにかくキータッチが心地よいことで，従来のこのサイズのポータブルワープロ(特にWD-300Fなど)と比較してみても，格段の進歩だといえそうだ。

これまでのこの価格帯ものは，どれをとってみてもあのPC-88mkⅡに勝るとも劣らずのキーボード感覚で，いつでもどこでも手軽に入力できるといったポータブルワープロの大前提が，１行入力しないうちに覆されて，最後には「おまえなんか仕方がないから使ってやっているんだぞ」と思わず叫びたくなってしまいそうになったことが思い出される。

まずこれでポータブルワープロとしての第一関門は無事通過である。次に16ドット通常表示(このほかに24ドット30字×８行，12ドット60字×15行の縮小表示/拡大表示も可能）で45字×13行の大型チルト式ＳＴＮ液晶ディスプレイ，これもまず見やすく視野角が広いので照明との角度や自分のポジションなどはあまり気にする必要もない。これも合格点をあげられそう。次に「変換効率を上げた AI 辞書搭載の新文節変換」というやつであるが，ここでいう AI 辞書を簡単に説明すると，ことばとことばのつながりを熟語形式で記憶していて判断し，変換してくれる辞書ということである。実用例をあげると，「厚い/友情」，「暑い/日差し」，「熱い/思い」というように，区切られた前後の文節の意味によってそれぞれ適切な変換語句を選び出して変換してくれる辞書ということらしい。

これは新文節変換と明示されていることからわかるように，各個人の使用状況下で実際に自分で使いこなしてみないことにはその善し悪しは判断できないと思う。ここでひとつだけいわせてもらうとすれば，先述のWD-300Fなどと比較してみると変換的中率ははるかに向上しているし，私個人としてはまったく辞書を学習させていない最初の状況からでも十分使いやすいと感じたのは事実である。ただし，AI 辞書だからあれもこれもと同音異義語を並べ立てて「さあ，変換してみろ」とやってみると，その結果は見事に裏切られてしまった。しかし，この辞書のネーミングはさておき，発想的には複雑な表現が必要とされる日本語処理において，大いに実用に向けて今後レベルアップしてほしいと思う部分ではある。

編集機能がさらに充実

入力においては，ひらがなをカタカナに変換する，またその逆を行う場合において，これまでは比較的最初にどちらかに設定してしまえばその後の変更ができなかったものと違って，入力後に［機能１］＋［かな/カナ］キーを使えばひらがな，カタカナのどちらでも入力したあとでの変更ができるようになった。これは全角文字を半角文字に変更する，また半角文字を全角文字に変更する場合も［機能２］＋［半換］，［機能２］＋［全換］キーを押せば入力後でも変更がきく。さらにはローマ字入力時にうっかり英数モードで入力したとしても［機能１］＋［かな/カナ］でひらがなに直すことができ，漢字に変換できる。このような作業はパソコン上でワープロを使っていれば当然のように思われるかもしれないが，この260Fと同時期に発売された最新機種を除いて，これまでの10～15万円台のワープロ専用機を使いづらいと感じさせていた最大の原因はこのような基本的な操作方法に起因していたわけだから，260Fを使ってみるとポータブルワープロの世界がようやくスタートラインに到達してきたような気がする。

編集機能のなかで，このほかに使いやすさといった意味で大幅に強化されているのが検索機能で，検索/置換/検索消去/短縮検索/書式検索/タブ検索の６種類の検索ができるようになっている。そのなかでもそれぞれの検索を行う場合には現在カーソルのある位置から文末方向，文頭方向の双方向への検索が可能となった。これは98用の新一太郎（ATOK5）などでもできなかったことで，このクラスのワープロにこのような機能が採用され始めたことには注目しておきたい。

ホームユースでの役割

このサイズのワープロ専用機を家庭内で使用するとき，「あってよかった」と実感させてくれるのが年賀状のシーズンである。これはまったく個人的な話になってしまうが，これまで使っていた WD-300F がわが家でもっとも活躍したのが年賀状，暑中見舞い，住所移転通知など案内状ハガキの原稿作成と印刷である。そんな使い方ではプリントゴッコと変わらないではないかといわれてしまいそうだが，300Ｆはミニ書院ＷＤシリーズで初めて3.5インチフロッピーディスクが搭載されたタイプだったためにデータの保存ができ，それをハガキにも印刷できることが有り難く感じたものだった。

それが今度の260Fでは，住所録の作成から，さらに全部宛名印字/検索宛名印字/宛名番号検索/郵便番号検索/住所検索/氏名検索/備考検索の７種類の検索印字機能を持っている。当然これは一種のデータベースとしての役割を果たしてくれるし，個人的に家庭内でのハガキ印刷（和文24×24ドット）だけを考えれば，これだけの機能を備えていれば十分だと思う。データベースだからといってこのほかにまだあれもこれもと望みたくなるのは当然かもしれないが，私個人としては電話番号や住所だけ知りたい場合などは，電卓サイズの電子メモひとつでいまのところはことが足りている。もっとも，その電子メモとこの260Fとのデータの互換性が望めるのであればそれにこしたこ

とはないのだが。

話を元に戻そう。ハガキ原稿の作成に関してさらにいえば，この260FにはオプションのハンドスキャナWD-01HS(29,800円)を使って，最大縦120×横60mmで写真やイラストの取り込みが可能であるし，イラスト作成機能も持っているので，その取り込んだ画像データをさらに編集/加工することもできる。こうなれば，私が家で文書作成以外に使うポータブルワープロの役割は十分にクリアしてくれる。ましてや，ユーザー辞書登録も見出し5文字，登録語3文字として最大約110件まで登録でき，実際には見出し語はひらがな，カタカナ，英数字で1見出し全角/半角20字まで，登録語は40字まで可能なので，ハガキなどの定型文書を作成する場合には最高の条件を備えているといえるだろう。

さらには書院カルクも搭載されているので演算，表集計，グラフ作成も簡単に処理することができる。これは次に紹介するWD-540の書院カルクと同じソフトを搭載しているため書院カルクについては次に登場するWD-540上で紹介することにしよう。

グラフ処理に威力を発揮

WD-540

このWD-540は，写真の外観からもわかるようにオールインワンタイプのWD-600シリーズの系統の機種で，基本機能はWD-260Fとほとんど変わらない。

ディスプレイは書院シリーズ当初から採用されている白画面黒文字表示（反転表示可能）で，9インチ(通常40字×15行表示)ながらも640×480ドットの鮮明な画像が実現されており，長時間の使用にも十分耐えうるだけの見やすさである。その9インチディスプレイをフルに使って画面を上下2分割できるマルチウィンドウ機能も備えている。このマルチウィンドウ機能はWD-260Fにも装備されていて，表組み作成時などにはその威力を発揮してくれるが，現状での表示行数でどこまで見やすく感じるか，またそれぞれ入力する文書の種類などによって大きく評価が分かれそうだ。

入力，変換に関しては260F同様に扱いやすいというのが第一印象だった。しかし，260Fを使ったあとですぐに540のキーボードを使ってみると，キーの配置が違っているのに気がつく。同じJIS配列準拠タイプのキーボードを使っているのに，機能キーはまだいいとしても，ひらがな/カタカナ/英数/取消の各キーのレイアウトがまったく違っているのには閉口させられた。当然これはマイナス要因である。

書院カルクについてはX1turbo用のMultiplanと操作はほぼ同じで，最初に作成する表/グラフの枠を指定し，そのあとでデータの表示形式と数値データ形式を決定する。そしてセルを移動させながら各データを入力していくわけだが，数値はもとより日本語入力においては昔からのカルクソフトの入力方法を見事に踏襲しており，相変わらず画面下で入力，設定，セル移動を繰り返しながら作表していかなければならない。カルクソフトがこの価格帯のワープロ専用機に搭載されたことは大いに評価できることだとは思うが，だからこそこのような手間のかかる入力作業から解放されることが当面の課題のような気がする。

そうして作成した表を今度はグラフ作成モードで処理すれば円/折れ線/棒/レーダーチャートなど11種類のグラフ作成をキー入力ひとつで行ってくれる。そしてグラフをそのまま印字することも可能であるし，職場で書院の上位機種を使っている場合などには，そのデータを利用することも可能である。これがWD-5010Dとなると，実際に自分の手で線や円はもちろんのこと扇形や楕円，放物線などを座標指定により簡単に作成できるようになる。

この540のそのほかの特長としては，260Fで紹介したことに加えて，JIS第1水準とともにJIS第2水準漢字（32ドット用）をROM化して内蔵していることや，罫線で指定した領域内で独立した文書の入力，編集ができるブロック編集機能を持っているため，文書作成においてレイアウト上簡単な割り付け処理もできる。さらにこのブロック編集したデータを移動し作成文中に流し込むような作業が可能になれば上出来だ。

540が5月に発売されたあと，今度は2ドライブ搭載し通信機能を持ったWD-640(188,000円)がこの7月に発売されたが，同タイプのマイナーチェンジのものを充実させるよりも，500シリーズでは編集機能が，また600シリーズではイラスト処理やカルクが充実しているなど，それぞれランク別の機種設定をポイントに置いて市場に送り出してもらわなければ，ユーザーとしてなにを基準に機種を選べばいいのか混乱を招いてしまうだけではないのか。ポータブルワープロの世界が広がっていくのは私としては大歓迎である。ただしそれはより合理的な進歩を望むのであって，“数撃てば当たる”式の増殖だけはご遠慮願いたい。

■ WD-540とハンドスキャナを使って作成した案内状(縮小率55%)

新しいマークを創りました。

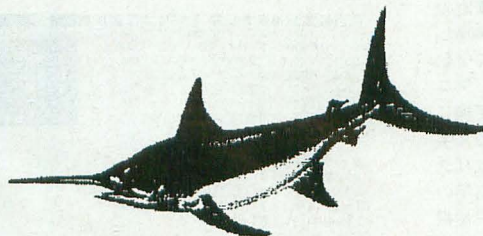
NEW DESIGN MARK

〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜 ♥ 〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜

もっと個性的で、温か味のある"便り"をしたい…。
こんな願いから、スナップ写真をもとに、新しいマークを創りました。
絵の仕上がりも鮮明で、満足しています。いかがですか？
これからは、楽しいイラストやスケッチ、地図などを
美しくデザインして、お届けしてまいります。

〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜 ♥ 〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜

この印字はシャープ「ミニ書院」WD－540で印字しました。

多彩な図形処理機能がポイント

WD-5010D

書院WD-5010Dは，先に紹介されているMZ-2861の書院28とは基本的に同じ機能を持ったワープロ専用機である。しかし，大きく違っているのは書院カルクと図形作図機能が装備されていることで，14インチディスプレイ上で思いどおりのグラフや図形を作成できることだ。

この5010Dのキーボードはワープロ専用のものとなっているため，機能キーやリピートキーが配置されているほか，カーソル移動キーがテンキーの下に付いており，また変換キーとスペースキーがMZ-2861のものと左右逆になっている。またMZ-2861での［ＳＦ］キーの役割が，［機能］＋［カタカナ］や［機能］＋［半換］などのキー操作に割り当てられているため，入力操作に関してはこちらのほうが楽である。

日本語入力などについては，書院28で後藤氏が詳しくレポートしているので，ここでは図形・グラフ処理についてのみ紹介することにする。

まずモードを図形かグラフに切り換えたあと枠設定を行い，図形の場合であれば，その次の図形の種類/線の種類/網の種類/図形を描く位置の４項目の設定を行う。ここで選べる図形の種類は，直線/矩形/多角形/円/円弧/扇形/楕円/楕円弧/放物線で，それぞれ座標をドット/ミリ単位で移動指定しながら書き込んでいく。さらに網かけの塗りつぶしを行ったあと図形データ上に文字を重ね書きすることもできる。

この場合，文字の表示方向を縦右/縦下/横右/横下/横上の５種類から選べて，文字サイズも普通/倍角/縦倍角/４倍角/小文字(16×16ドット)のなかから自由に指定できる。さらに文字を表示させるときに，図形上に単純な重ね書きと背景白抜き表示が選択できる。こうして書き上げた図形は，消去/移動/複写/縮小/拡大などの編集を行うことができ，さらには複数の図形の合成や位置指定による回転も行える。

この一連の操作は比較的簡単であるし，領域指定した枠のなかにそれぞれの形をした図形がどのような比率で書き込まれるかさえ把握してしまえば，自分の思いどおりのサイズの図形を描くことができるようになる。ただ悲しいことに個人的にワープロを使うにあたって，これまでZ'sWORDや花子などは使ってみたことはあるものの，実際にこのような図形作成機能を持った専用機というのは使った経験がないため，この機能が実務レベルでどうなのかを判断するまでには至らなかった。

グラフ作成に関してはまったく540と同じで，グラフモードから作表したあとは種類を選択してキーを押せばそれぞれ自動的にグラフを作成してくれる。

とにかくこの書院シリーズというのは，今回のように各シリーズごとに使い分けてみても，基本が同じなので操作性など極端な使いづらさというのは感じられないが，なぜか260Fもこの5010Dも同じで未確定の文中に訂正を加えて，カーソルがその範囲内を移動しているときに変換キーを押すと，カーソルから前の文節しか変換してくれない。これは１年半前からまったく進歩がない。辞書の強化，編集機能の充実などは，この期間になされて当然のはずである。それなのに5010Dにあっても日本語入力の基本部分が進歩していないのは残念だ。

ワープロ専用機として書院はますます進化してほしいと思う。いまでもこれだけタイプの揃った書院シリーズは，540のところでも述べたようにポイントとなる部分での改革を忘れたままファミリーだけが揃ってしまったという状況を早く脱皮して，小さく独立した個性的なファミリーの集まりに変わってほしいと思う。

Shogun，ついに出番です

ようやく“Shogun”が出た。といっても残念ながらまだ88版の話。以前，X1版Samuraiのレポートをお届けしたときに「来月はShogunのレポートです」と予告してしまってから幾年月。そのあとずっと姿を現さなかったこのソフト。これじゃあ名前は“忍者”の間違いではないかと思ったほど待たされていたのが，ようやく完成が近づいてきたらしくなんとか夏中にはお目見えとのこと。ここで基本機能が同じだということらしいので，88版2HDのShogunを使って顔見せ興行といってみましょう。

最初は，“即戦力”のお兄さんぐらいだろうの気持ちでマニュアルも満足に読まずに文書を作ろうとしたら，まず文書画面からまったく違うのにビックリ。これではいかん，とあわてて「15分間マニュアル」を読んでいざやってみようと試みる。

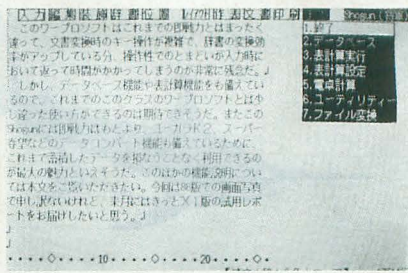
それぞれファンクションキーに入力/編集/装飾/辞書/位置/レイアウト/作表/文書/印刷/ほかの10個のメニューが表示されていて，選択するとプルダウンメニューが開かれる。「こりゃ便利」と喜んでさっそく入力モードから「連文節変換」を選択してやってみることにする。

まずはマニュアルに書いてあるとおりに「かがくをまなぶ」と入力して変換キー（スペース）を押す。すると「科学を/学ぶ」と緑に白の反転文字と赤に白の反転文字が文節で分れて表示された。そこでスペースキーを押すと「1：科学を　2：化学を　3：価額を　4：かがくを　5：カガクを」と５つの変換候補が画面下に表示され，さらに変換キーを押していくとしだいに候補の上をカーソルが移動する。しかしその段階では文中に表示されている文字は変わらない。どうやらリターンキーで初めて候補を画面上に呼び出して［↓］キーで確定する方法らしい。これは一太郎を意識して作られたのかもしれないが，未確定の文節を切り直す場合などは［ESC］キーで解除して［←］，［→］キーで文節を切り直すことになる。

とにかく慣れるまではたいへんである。マニュアルは例にもれずわかりづらいし，重変換しようと思えば青色反転/緑色反転/赤色反転文字が画面上で飛び交うことになる。さらにはリターンキーで候補を確定する方式なので，右手の指は「ああでもない，こうでもない」とキーを探してうろうろすることになる。

その変換の手順を除いて辞書は扱いやすいようだ。比較的自分が変換したい文字を最初に表示してくれる確率は高い。そのほかにこのShogunには，カード型データベース機能や表計算，それに2D/2HD版それぞれにおいて各機種の即戦力，JET，ユーカラK2，Super 春望で作成したデータのファイルコンバートが可能なほか，88MR/MH用の2HD版においてはMS-DOSの文字ファイル（テキストファイル）データも取り込める。

とにかく今回はあわただしく，ただただShogunとはいったいどんな顔をしたソフトなのかの概要を追っかけるだけで，評価どころの騒ぎではなかったことをおわびしたい。いずれX1turbo版でその真価を問いたいと思うのでお楽しみに。

特集 迷宮の日本語処理環境

私のワープロは常に最強だ

自分だけが楽しく使える専用ワープロがほしい，これは誰しもが考えてしまうこと。そこでOh!MZのスタッフに夢と現実，その他いろいろ織り交ぜた独自のワープロ観を語ってもらおう。

4つのお願い聞てよね

Shimizu Kazuto 清水 和人

たとえば私がワープロに恋をするなら，4つのお願い聞いてほしいの(ちあきなおみ)。

ひとつ　でたほいの文章変換

まずはなんといっても文章変換，次のような機能がほしいと思う今日このごろである（夢にまでは見ないが……)。

1)　意味解析自動変換

文章の内容を理解したうえでの変換だから，誤変換が少ない。たとえば次のような文章も一発で変換する。

「ハシクンハハシノハシヲハシッテハシヲハッシトウケトメタ」

「橋君は橋の端を走って箸をハッシと受けとめた」

うーん，こんなの人間でも間違えそうだ。人工知能の世界かもしれない。だけどこれが第一歩，あとの要望はもっと大変!?

2)　文体一括変換

ワープロで打つ文章にもいろんな目的がある。その目的が異なれば文体だって違ってくるはずだ。次の例を見てくれたまえ。

「きょうは雨だ」

「本日は雨でございます」

「なに，きょう雨じゃん」

このような文体の一括変換が可能である。書き手の気分によって「うんざり調」，「怒り調」，「喜び調」などが，また読み手の層によって「園児調」，「奥様調」，「猪木調」などが選択可能。同じ内容でも文体によって全体の雰囲気がガラリッと変わる。

「きょう雨かよお」うんざり調。

「なにい？　雨だあ？」怒り調。

「わぁい，雨だ，雨だ」喜び調。

「あのね，あめがふってゆの」園児調。

「あーら雨ざーますわ，オホホホ」奥様調。

「雨だ，このやろ」猪木調。

つ，使える！

3)　外国語一括変換

もちろん外国語へも一発変換するのだ。英語，仏語，露語，中国語……。うーんでも例文がわからん。便利なハズなんだが。

‘It’s rainy today.’

うーん自動翻訳はまだまだ先だというけど，なんかおあずけをくってるみたいでいやだ。早くパーソナルユースにしてけろ。

4)　方言一括変換

これは便利！　日本人の心のふるさと，それは「方言」。ならば日本語ワープロには必須なのが方言変換。次の例文を見よ。

「何をおっしゃっているのですか？　私にはよくわかりませんが」

「なにゆうてけつかんねん。いっぺんしばいたろか，このだあほ」

ふたつ　ふらちな入力方法

入力方法だってまだ現状ではまだるっこしい。以下のような機能を待ちこがれる。

1)　音声入力

マイクに向かってしゃべればどんどん入力，自動変換と合わせればなかなか快適。もちろんTVのニュースやレコードの歌詞がそのまま文字になってくれる。これならもうキーボードなんていらない。文法や用語のミスも的確に修正してくれる。

2)　手書き，印刷物入力

紙の上の文字をそのままデータに。人間に読めるんだ，コンピュータに読めないわけがない。汚い原稿から新聞の活字まで手軽に入力，文体一括変換で大作家の文学を文体変換できちゃう。うーんOCR。

「私は猫でございます」(夏目漱石)

3)　リアルタイムペン入力

特殊プレートの上に書いていくとそのまま画面に活字となって現れる。消しゴムも使えるし，落書きやイラストがそのまま入力できる。うーんグラフィック一体型。

みいっつ　みだらな印刷機能

次は印刷機能だが，私は次の要求をかかげて断固闘うぞー。

1)　紙上レイアウト機能

画面上に印刷イメージが出る。マウスでピックしてこの部分をこっちへ，あれをそっちへ，ってんで自由かつ気ままなレイアウト作業。フフフンフンフンついでにBGMも流して高級感を出そう。

2)　高速印刷

おらおら，どいたどいたあ。車に当たっても知らねえよ！　それはもう暴走気分，100枚だって1000枚だってわずか1分，まさに印刷所気分。OCRと組み合わせてニセ札だってひょひょいのひょいっと。

3)　豊富なフォント

「明朝・ゴシックはあたりまえー」

「当たり前のこといわないの！」

少女っぽい丸文字から文豪っぽい達筆，自分の字体を「字体クリエイター」でエディット，もう君の文書をワープロだと思う人はいない。「最初はていねいに，しだいにくずして」などという「清水和人原稿調」も選択可能である。

よおっつ　たいじてくれよう音声出力

へへへえ，次のような機能もほしいのさ。

1)　朗読機能

できた文章を印刷するだけではおしい。文体変換で演説調にしておけば政治家顔負けの大演説ができる。もちろん声は「田中角栄」，「三船敏郎」，「田原俊彦」などといったメニューからも選べる。外国語変換と合わせれば，全世界に物申すことだってできる。印刷物入力で台本の各自のセリフにいろんな人の声を登録すれば，あっという間に「ラジオドラマ劇場」のでき上がりってもんだ。

2)　歌唱機能

うーん，付属のミュージックエディタで作った楽譜に合わせて歌ってくれるってとこかな。クラシック調，アイドル調，なんでもござれ。

えーっ，なんだってえ，こんなの全然ムリだってえ？　へん，で，できなきゃいっちゃいけないのかよ？　い，いいもん，タコにはタコの生き方があるんだもん。といいつつ，直径2cmほどの石ころを1.5mくらいずつけっぽって夜道を去って行く私であった。

遥かなる日本語への想い

Izumi Daisuke 泉　大介

漢字と仮名をとりまぜて表記する日本語。幼いころから当然のように使ってきた日本語を，美しいと思ったのは高校生のころだと思います。詩を書くようになって私はますますこの言語の魅力に取り付かれていきます。速い，疾い。口語では同じように使うこれらの言葉は，文字にするとこんなにもイメージが変わってきます。漢字は表音文字ではなく表意文字なのです。「速い」ではなく「疾い」こそがこの状態を表現し得る漢字だ。そういう選択が可能な言葉なのです。同じように「はやい」と読んでも，そこに込められる意味には大きな違いがあるのです。

つい最近のことです。勝本信氏の「Between The Lines」を読んで私はたいへんショックを受けました。漢字ROMに登録されている「はるかかなた」というときの「はるか」という字が自分の覚えていた字と異なっていると書いてあったのです。あわててMZ-2500を立ち上げて変換してみると，確かに「遥」という字が出てくるではありませんか。第2水準のROMのほうには古い書体が入っているとも書かれていたのでシンニュウのところを捜してみました。しかし「遙」という字はありません。あわてて第2水準のすべての文字をかたっぱしから見ていくと，やっとJISコード7423のところに見つかりました。

私が「遙か」という字を覚えたのは中学生のころでしたから，もう随分と昔のことになります。それ以来「はるか」と書くときには必ず「遙か」という字を使ってきました。いきなりそこへ「遥か」という字を見せられたのですからそのショックがいかに大きかったか想像していただけると思います。どうしてこの字が「はるか」という意味を表せるのか理解できなかったのです。そこにどんな空間的な拡がりをも感じ取ることができなかったのです。「遥か」はもはや漢字ではありませんでした。それは単なる約束事に過ぎません。「遥か」と書いたら「遙か」という意味を表すことにしよう。ただそれだけのものとしか私の目には映らなかったのです。

いったい漢字を簡略化することにどんな意味があるのでしょう。今後ますますワープロは普及していくでしょう。それに伴って「書くのが面倒だから簡略化する」という理由はまったくの過去の遺物と化していきます。本来意味を表す表意文字である漢字を，簡略化によって単なる約束事，ひいては単なる記号と化してしまうことにいったいどんな意味があるというのでしょう。簡略化という名目で，然るべきツクリを別のものにすり替えてしまうことにどんな意義があるというのでしょう。日本語はすでに十分簡略化されていると思います。もうこれ以上の無益な簡略化はやめてほしい。心からそう思います。

漢字と日本語

漢字には非常に多くの表現方法(文字)があるにもかかわらず，日本語にはそれに対応する言葉がないために同じ読みを当ててしまったという場合が少なくありません。逆にいえば日本語では同じ言葉をさまざまなニュアンスで用いるということになります。われわれは言葉だけではなく，会話の流れからそれがどういうニュアンスで用いられているのかを理解しています。

しかしながら詩文においては意味を正しく読み手に伝えようとして，現代の日本語ではあまり気を使わず，ひらがなで書いたり一般に使われる漢字で済ませるところでも漢字を取捨選択することがあります。詩文の場合はそこに使われている言葉がどういう意味なのかを判断できるほど前後関係を説明しないためでもあるでしょう。文ではなく言葉で意味を伝えようとするからです。簡単なところでは「おもう」という漢字に「思う」ではなく「想う」を採用したり，あるいは「恩う」を採用したりということがあります。漢和辞典のなかには巻末に日本語の読みに対する漢字がその説明付きで掲載されているものがあり，正しく意味を伝えようとする場合には重宝します。

ワープロで詩を書こうとする場合には，以上述べてきたような点が問題になり，使うワープロによっては無謀な企てともなってしまいます。まず，漢字ROMを採用し変換の高速化を図った結果，ユーザー登録ができなくなってしまったものは論外です。「おもう」を変換して「思う」と「主う」，「重う」などが出てきた日には目も当てられません。仮にユーザー登録できたとしても，ユーザー登録語を呼び出すキーが通常の変換キーではなくファンクションキーであるならやはりこれもご免被りたいところです。

ユーザー登録ができ，最初から入っている漢字と同じように変換キーによって変換できる場合でも，登録が品詞分類して行われないものならこれも意味がありません。「想う」という単語は常に「おもう」と使われるとは限らないからです。「想っている」，「想い」などを入力しようとするときに，まず「おもう」を変換して「想」を出し，その後ろに「っている」を付け加えなければならないというのは地獄です。入力のリズム，感性のリズム，心のリズムを著しく損なってしまいます。文を書いているのではなく，文字を組み合わせているような感覚に陥ってしまうのです。

このような条件をクリアしたワープロを使い詩を書き始めても，最初のうちは辞書に使いたい漢字が登録されていないために苦労することになります。またある程度語彙が整ったあとでも，新しい表現をしようと思えば再び漢和辞典とユーザー辞書登録のお世話にならなければなりません。

そもそも日本語ワープロというのは日本語で文章を作るものではないのでしょうか。文章をわれわれが作るときにはペンと紙と辞書を手元に置いて書きます。確かに日本語ワープロはペンと紙の役割は果たしてくれます。しかし辞書についてはまったくです。英文ワープロのなかには難しい単語を使うと「こっちのほうが一般的だけど，このままいく？」ってなことを尋ねてくるものがあるそうです。それでこそ本当のワープロなのではないのでしょうか（もちろんこの機能がON/OFFの選択のできないものであったなら，それは単なる「小さな親切大きなお世話」になってしまう)。

私は特に漢字にこだわる人間です。しかし，この言葉を表す的確な漢字はないものだろうかと思った際にいちいち漢和辞典を引かなければならないというのは，非常に面倒なことなのです。ワープロを使っている最中に内蔵の漢和辞典に照会できればど

んなに便利でしょう。辞書を引くといういちばん面倒な作業はコンピュータが受け持ってくれ，私たちは表示された意味を見ながら使う漢字を選択できるのです。そしてやっとワープロはペンと紙と辞書の代わりになるのではないでしょうか。

文字送りと行送り

手書きで詩を書いている間は意識していなかったのに，ワープロを使い始めてから気にするようになったことのひとつに，文字送りがあります。文節と文節を全角のスペースで区切るほどではないし，かといって続けてしまいたくもない。ここは半角のスペースにしておこう。といったことです。

行送りのほうも同じです。1行空けてしまうほど意味が離れてはいない。かといってまったく空けないというのではイメージが伝わらない。半分だけ改行してほしいのだが，ということです。残念ながら印字するときの改行幅は設定できても画面上にそのまま表示してくれるワープロは少ないようです。印字イメージをそのまま画面に表示してくれ，しかもその状態で行送りをマウス等で簡単に設定できるようになっていれば好都合なのですが。私は言葉によるイメージだけではなく詩全体の視覚的イメージも大切にしたいのです。

全体の改行ピッチも大きな問題となります。意味の都合上どうしてもここまでは同じページに収めたいということがあります。改行ピッチを変えることができないと，ひと続きの詩がページによって分断されてしまうのです。ページを越えて目を運ぶことによってそれまで創造してきたイメージが大きく損なわれてしまうのです。

昔，私の書いた詩を学園内の新聞に載せたいという申し出があったときに，ここは半行送ってくれ，ここは1ページに収めてくれと，編集をやっていた同学年のG君をさんざん煩わせたことを思い出します。

詩とは考えや意見以上にイメージを伝えるものだと私は考えています。心の中に描き出される漠然とした想いを，ある出来事を創造することによって，疑似的に伝えようとするものだと思います。心の中のイメージは自由なものです。それが詩という形態をとって描き出されたときに，ワープロの機能によって束縛されるようなことがあってはいけないのです。

自由な想いを自由に書き綴ることができる。私がワープロに求めたいのはこの1点です。

忘れないでほしいワードパワーの発想

私は，常々日本語ワープロというものに疑問を持っていました。日本語ワープロというものの一般的な定義はどのようなものかというと，日本語を入力して，それを編集し印刷するものである，といったところでしょう。

優秀なワープロというのは，優秀な漢字変換機能を持つフロントプロセッサと，優秀なエディタと，優秀な印字機能があればいいのです。ビジネス用ならいざ知らず，パソコンを使ったパーソナルユースでは，本来文章を書くという作業はクリエイティブであるはずです。果たして，先のような定義における日本語ワープロというものは，このクリエイティブな作業にとってどの程度の手助けをしてくれるのでしょうか。なるほど，ワープロを使うと楽に文書が書けるようになります。実際，この原稿もワープロで書いています。しかし，これは本来であれば紙と鉛筆を使った作業が，ただパソコンとプリンタに置き換わっただけにすぎません。肝心要の文章は，人間がすべて頭のなかで考えなければいけないのです。人間がなんらかの構想を練る助けをするアイデアプロセッサのようなソフトもありますが，日本語ワープロはほとんど人間の知的作業の分野には入り込んではきません。考えてみてください，「こいつを使ったら面白い文章が書けそうだな」などと思わせるような日本語ワープロが果たしてあったでしょうか。文章を書くということに関しては，人間の知的作業の手助けをするどころか，人間の感性を刺激してくれることさえもできないのです。

たとえば，PC-9801やX1シリーズでお馴染みのグラフィックツール「Z'sSTAFF」について考えてみてください。このソフトの優れた点は，さまざまな編集機能・特殊な効果によって，「絵を描く」という人間のクリエイティブな作業の一部を肩代わりしてくれます。人間の感性を刺激して，「このソフトを使えば，こんな絵ができるかもしれない」と人間に思わせるのです。こんなグラフィックツールとは違って，いわゆる「日本語ワープロ」なるものがこの先どんなに進歩したとしても，このままでは字を書くための単純な道具の域を超えることはできません。

むかしむかし，X1turboII なる機種があったのを覚えておられるでしょうか（現在持っている人はごめんなさい。そんな昔の話じゃないですね)。従来のX1turboを低価格にしたうえに日本語を強化し，初のブラックカラーで発売された機種でした。このX1turboIIの日本語処理の目玉は，なんといってもワードパワーであります。これは，簡単にいってしまえばかな漢字変換用の辞書なのですが，単なる辞書を超えた新しい発想のもとに作られています。かなを入れて変換すると，ただその読みの漢字が出るだけではありません。類義語，反意語まで出てしまうのです（MZ-5500でもこの機能が話題になりました)。反意語はともかく，類義語が出る辞書というのは非常に興味深い辞書であるといえます。

本来，辞書というものは，言葉の意味・書き方を読みから引くやり方が普通です。ところが，このワードパワーは簡単な意味を入力することによって，その類義語を調べることで新しい言葉を探すことができます。まさに辞書とは逆の発想なのです。ビジネス用の文章を書くのではなく，純粋に個人的な文章を書くのであれば，できるだけ気の利いた文章を書きたいと思うでしょう。しかし，自分の知識のなかの限られた語句内で文書を書いていてはどうしても文章が型にはまったようになりがちです。そんなときにこのワードパワーが使えれば，類義語などのちょっと違った言葉を調べて使うことによって，文章を豊かにすることが可能になるのです。

ワードパワーのもうひとつの特徴は，かな漢字変換辞書のデータベース化です。このアイデア自体はそう珍しいものではないのですが，実際にこのワードパワーを使ってみるとそのよさがよくわかります。そのなかでも文章の作成に役立つものとしては，手紙によく使う言葉・用例があります。私は実際に，簡単なエディタのようなものを作って手紙を書いてみましたが，ほとんど頭を悩ませることなく書き上げることができました。手紙などの場合は目的を持ったものが多いので，それにふさわしい語句というものは前もってある程度は予想できます。いちいちその場で考えるよりも，そのような語句をあらかじめ用意しておいて，そのなかから選ぶことによって考える手間が大いに省けるはずなのです。

このように，ワードパワーを使えば紙と鉛筆だけで書くよりはひと味違った文章が書けるのではないかと思わせてくれます。しかし，実際にワードパワーを使って文章を書いた人はあまりいないでしょう。いかにワードパワーが優秀でも，それを生かすエディタなどがまったくといっていいほどなかったからです。ソフトハウスがそのようなエディタを作ってくれなかったのは仕方ないとしても，メーカーのサポートがなかったのは残念でなりません。さらにメーカーがサポートしてくれるどころか，次に発売されたX1turboIIIではせっかくディスクの容量が増えて使いやすくなったのにもかかわらず，ワードパワーは付いていませんでした。

このワードパワーが出た当時は，まだいわゆる「日本語ワープロ」の平均的な性能もまだ低い水準でした。いかにして正確なかな漢字変換を可能にするか，いかに優秀な編集機能を作るかなどに重点が置かれていました。その点を考慮すると，少しばかりワードパワーの出現は早すぎたような気もします。しかし現在は，いわゆる「日本語ワープロ」として機能的にはほぼ十分なところまできているような気がします。もしこれが本当に十分なところまで行ってしまうと，次は，ワードパワーのような考え方が日本語ワープロに取り入れられてくることでしょう。もしそうなったときには，実績のあるシャープさんにぜひともその実力を発揮してほしいものです。　（山村　一）

日本語は芸術するのだ

Yoshida Kouichi 吉田 幸一

どうも中学生というのは作家になりたいと思う年ごろらしい（特にコバルト文庫好きの女の子)。私もその例に漏れず，SF作家になりたいなどと夢見てノートや原稿用紙に殴り書きしては自己満足に浸ったり，クラスに回覧したりしたものである。たいていの人間は高校へ入り，思春期を越え，大人たちの論理に巻き込まれ始めるころには単なる普通の“微熱少年”と落ちぶれるのだが，大人になれない私は未だに夢を捨て切れない始末である。

日本語は芸術である。文学は芸術である，という意味ではない。確かに文学は，音楽や絵と同様芸術たりうるが，それはここで述べるべき問題ではない。主張したいのは，日本語で書く言葉，文章それ自体が芸術であるということだ。その秘密はいまの日本語が3種の文字を持つ点にある。ひらがな，カタカナ，漢字である。この3種の微妙なバランスによる綾は口に出しただけでは表現できない。それこそ書いて初めて伝わるものだ。

まったく同じ文章でも，3種のうちのどれを使って表現するかでいかにニュアンスが変わるか。

“はるかな迷路のひだを通り抜けて，とうとうおまえがやって来た。”

これは安部公房『他人の顔』の冒頭である。特に漢字が多いわけでもない普通の文である。無駄のない，実にいい文である。

“遙かな迷路の襞を通り抜けて，とうとうお前がやって来た。”

これが漢字多用版。

“はるかな迷路のヒダを通り抜けて，とうとうオマエがやってきた。”

これがカタカナ版。

3つの差は歴然である。いくらこのあとを読まなければわからないといったって，2番目のは文の前半分に漢字が偏って重くなり，アンバランスな印象を与えるし，最後のはセンスのない格好だけつけたおかしな文となっているのは改めて説明するまでもないこと。

この3つを一般的，文学風，イマ風などと十杷ひとからげ的に分類してしまってはいけないのだ。見た目の違いだけで同じ文章をこうも書き分けられるのは日本語だけである。

“僕”と書くか“ぼく”にするか“ボク”がいいか。河と書くか川と書くか。同じ意味で同じ読みの言葉を束ねても美しくなったり見苦しくなったりさまざまな姿を見せるのは日本語だけである。

日本語処理というからには，作る側も使う側もそこを肝に銘じておかねばならない。手で書こうが機械に書かせようが，文章は書き手の表現なのである。

というしだいで，やっとワープロの話が始まる。ワープロを使い出すとその人の文が変わるとか，定型的になるとか，没個性的になるなどという人がいるがそんな意見はクソ食らえである。書きたいのに書き方を忘れた字を書くのにいちいち国語辞典を引かなくてもよいというだけで，かなり自由に文章が書けるはずであり，またそうならなければ嘘なのだ。ワープロによって“読める字は書ける”時代が来る（べき）なのである。

遺憾ながら，そこまで環境は整っていないし，人々の意識もそこまで機械を受け入れていない。ユーザーに強制するワープロ，ワープロに使われるユーザー。責任はどちらにもある。言葉にこだわりたい作家志望青年としては名文を書けるワープロ，書き手のセンスを生かすワープロが欲しいのだ。気の利いた言葉を書くのに国語辞典を必要としなければならないワープロなんていらないのである。

人類皆小説家とはいわないが，誰でも一生に何回かはこれはという物語を創り上げるものであり，自分だけの言葉を持つものである。そんなときにそれを表現できる道具がないとしたら，とても悲しいではないか。

日本語処理とは呼んでやんない

とにかく，いまのワープロは志が低いというか，開発コンセプトが狭量で粗野で軽薄である。誰もが日本語を使いこなせるよう存在するべきなのに，幅広くて奥深い日本語の枠を自ら狭めているものばかりだ。ビジネス文書，特に1ページ1ページ完全に分割できるような文書がとりあえず作成できればいいと思っているのか。ビジネス文書に強いだけがワープロではないはずだ。そんなワープロはビジネスパソコンにだけ載せればよろしい。

「記者が汽車で帰社した」なんて文はどうでもいいから，たとえば新井素子節を変換できるフロントプロセッサを作ってみなさい。うまく変換できなかったからといって，せっかく思い付いた気の利いた文章をビジネス文書文体ですませてしまう，そんな気にさせる変換機能など言語道断である。誰でもが持っている自分の文体を機械に合わせて変えてしまうなんて，日本語処理以前の由々しき問題だ。

特に小説を書いていると，地の文ではあまりムチャしないので問題はないのだが，登場人物が会話を始めると途端にイライラし始める。連文節変換だろうが辞書先読み自動変換だろうがただの大きなお世話になってしまうのである。いや，連文節変換だからこそそうなってしまう。カタカナの専門用語や固有名詞の多い文章のときも同様である。どのワープロも好き勝手な字に変換してくださるのだ。もうあきれて笑う気にもなれない。泣きながら修正するのは嫌である。せめて，文法的に解釈不能な言葉が出てきたらひらがなのまま残しておいてくれるくらいの気遣いはあって当然のことではないか。

いくら辞書を整えても，辞書にある語を入力したときとない語を入力したときのギャップがあまりに大きくては，機械に合わせた文章を書くようになってもユーザーを責められまい。私だってつい気を抜くと，あまり出てこない登場人物には人名辞書にある名前を付けて楽をしてしまう。こんなことではいけないのだ。自分だけの文章を書くためには際限ない辞書登録を賽の河原の石積みのように繰り返さねばならないのだろうか。

話が後ろ向きになってきたので回れ右をしよう。

日本語処理と呼べるもののために

誰にでも書きたい文章が書けるワープロ，その人の能力を最大限以上に引き出せるワープロ，文才を磨けるワープロが欲しい。

とりあえず余計な機能はどうでもよい。

まずは感覚的で，無駄な動きと神経を使わない入力である。手書き入力なんていらない。手で書かなくてもすむからワープロである。音声入力なんてものもあるが，誰に向かってでもなくボソボソと喋るなんてぞっとしない。不気味なだけである。もう少し工夫をしてくれればキーボードで十分。TRONキーボードだろうが親指シフトだろうがいまは亡きM式だろうが（実は昔M式キーボードの隠れファンだったことがある）JISキーボードだろうが趣味の問題でどれでも構わない。片手で入力できればいいが贅沢はいわない。問題は変換動作である。ATOK（一太郎のフロントプロセッサ）のように下向きカーソルで候補確定なんて冗談じゃない。

操作性がよくても中身がボケていたら腐った林檎である。日本語ほど千差万別十人十色の文体を持つ言語はあるまい。すべてをサポートしろなんて無理な話だが，文体の学習機能や文体登録等の技があってもいい。かのワードパワーのような類義語辞書やことわざ辞書も当然である。

手軽な入力のために，いつでもどこでも寝転んでても使えるキーボードとディスプレイだけのポータブル端末もつなげたい。

ともかく，日本語を喋るように書けることがなによりも重要であり，そのためにワープロはあるべきなのだ。そうなると，書くことに無縁だと思っていた人々にも新しい世界を提供できる。

日本語は芸術である。誰でも芸術家である。試しにいまベストセラーとなっている俵万智さんの短歌集『サラダ記念日』を読んでみるといい。そこには特別な感性も難しい解説も不要である。既成観念に捕らわれた人はぶっ飛ぶだろうが，これが短歌であり自由な日本語なのだ。

書くことの楽しさを，貴方に。

フロントプロセッサ文法テスト

日本語処理でメインとなる変換機能は，フロントプロセッサと辞書とに分けられる。フロントプロセッサは入力された文字を解析して品詞などを決定し，辞書から適当な単語を捜し出して表示する。

動詞や形容詞など述語は変化する。名詞には助詞が付く。フロントプロセッサはその判断をせねばならない。よって，誰でも中学・高校で悩まされただろう国文法が役に立つ，どころか欠かせないのである。

ワープロに最も関係のある文法はまず品詞。品詞についての情報がきちんとしていないとアクロバット変換を平気でしてしまう。特に活用をする用言が大事である。

では，中学生用文法の参考書を武器にX68000に勝負を挑んでみよう。

ここでは——動詞・形容詞・形容動詞・名詞・副詞・連体詞・接続詞・感動詞・助動詞・助詞——と10種類ある品詞のうち，活用をする（つまり語尾が状況によって変化する）品詞の代表，動詞と形容詞・形容動詞の検査をした。

1） 動詞の変化

五段活用・上一段活用・下一段活用・カ行変格活用・サ行変格活用——すべての動詞についても調べたが ——未然・連用・終止・連体・仮定・命令——どの活用形でも（文法的におかしなところで使わない限り）まっとうに変換してくれた。当然である。

それでは，ということで，音便に挑戦。五段活用の連用形にしかないという音便は3つ。イ音便，撥音便，促音便である。“聞いて”は“聞きて”のイ音便，“読んで”は“読みて”の撥音便，“打って”は“打ちて”の促音便。少しは昔（あるいは最近）学校で習ったことを思い出しただろうか。

こちらも一発で合格。しかし，元のかたち（音便形でないもの）については不合格であった。いまどき“聞きて”などという人はいないだろうが，古い言い回し（古文では音便形はない）が好きな人にとっては迷惑な話。古典的な言い回しはほとんどないと考えてよいようである。

2） 形容詞，形容動詞の変化

“寒い”とか“恐い”など終止形が“～い”となるものを形容詞，“静かだ”や“あさはかだ”など終止形が“～だ”となるものを形容動詞と呼ぶのだが，これらの活用も基本なので，すべて合格，きちんと変換してくれた。

続いて形容詞のウ音便である。“寒うございます”の“う”がそうであるが，まったく駄目，不合格である。確かにあまり使うとも思われないし，“ございます”や“存じます”の前にしか付かないのだが“ございます”を辞書に持っているのなら，ないとおかしい。

句読点変換はこうして滅びた

日本語には文の切れ目に入れる句点，読点というものが存在します。この発想は句読点変換機能が付いていれば，文章を入力中にこれを入れる度に，それ以前の文節が漢字変換されるのでこれは便利，といって素直に喜んでいられる状況でした。

ところが実際は，ワープロのキーボードに慣れないうちはゆっくりと入力された文字や文章を確認しながら打ち込んでいくからいいのですが，そのうち慣れてくると人間というものは面白いもので，

1） 自分のキー操作に自信が出てくる
2） 気持ちが先走って，ディスプレイを満足に見ない
3） ミスタイプはあとで確認すればいいと思い始める
4） 句読点変換機能の存在を忘れる
5） 文中の句読点入力を意識しない
6） ディスプレイ上の，あるいは打ち出した文章を眺めては「ワープロとはなんてバカな機械なんだ」と責任転嫁する

といったごく一般家庭にありがちな人間模様が展開されるのです。実際，句読点変換機能の付いたワープロを使って実例を挙げると，

「私は，昨日，公園で，買ったばかりのビデオカメラを使って，風景を映していました。」というような文章を，そこそこワープロが使えて，ややおちょっこちょいの性格の人に一気に打ち込んでもらうと，

「私は，機能，公園出，買ったばかりのビデオカメラを使っ手，風景を移していまし田。」といった結果を招いてしまうのです（この変換実行結果は実在のものです。ただし使う機種によっては文字に若干の違いが生じますのでご注意ください）。

この句読点変換というのは，入力する人間の思考の切れ目，切れ目で文字を変換しようという，実に健全な発想だったとは思います。しかし，文章を無意識に入力するようになればなるほど，入力し終わったあとで句読点の存在する箇所をすべて点検し直さなければならないということにあとで気づかされるのです。

これは最近のワープロの標準装備になりつつある自動変換，一括変換などにも一部いえることですが，これらの変換機能というのは定型文書のみを打ち込む場合は非常に効率よく変換を行ってくれても，フォーマットの決まった文章の1カ所にでも手を加えると，とたんに効率の悪いワープロへと変貌する可能性が高いのです。ましてや，仕事以外にワープロを使っている場合ではなおさらのことでしょう。

変換機能はどれが最適か，それは使っている人によってそれぞれ評価が違っていることだと思います。しかし「発想はよかったけれど，実際はねぇー」といったレベルのアイデアだけで実用性の伴わないような変換機能は，それはワープロとして現実に存在する以前にすでに失敗作なのです。完璧な日本語処理をワープロに望む，それは現状では非常に困難なことだとは思います。それでもユーザーのどん欲さというものは，常にブラックホールと同じようなものなんです。これからの変換機能の充実にぜひ期待したいものです。

（H.N.）

また，この参考書には感動を込めて語幹だけで使われる場合として“おお，寒”というのがあった。形容詞は語幹だけで変換できるか，という問題である。結果——不合格。困ったものである。

ちなみに，MZ-2500の辞書ROMでは語幹だけでも変換できた。

3) おまけ

しかし，逆に文法的に間違っているのに変換できたものもある。“着れる”である。実は“着られる”（動詞“着る”＋可能の助動詞“られる”）が正しいのだ。これが間違っているなんてもう誰も知らないようだし，いまとなっては文法のほうを手直しするべきなのだろう。ちなみに，“着れる”はMZ-2500では変換できなかった。

というわけで，今回はこれだけである。日本語文法がいかに複雑でうっとうしいものか実感したが，ここでそれを示すにはページ数が足りないのでよしておこう。いうなれば，文法より先に言葉があったのだ。それをこじつけた理屈が文法なのである。初めに言葉ありき，である。

文節変換のみの時代なら文法など適当でも腕力でなんとかなっただろうが，2文節最長一致連文節一括変換などというややこしい複雑な変換が採用されるとそうはいかない。フロントプロセッサが自分で1文節の長さを判断したり，前後の入力から単語を判断するので，確かに普通の文章ではよいかもしれないが，逆に文法にそぐわない文を書いたり，フロントプロセッサ側の文法解析が甘かったりするとあられもない漢字を吐き出してしまう。

いまは，完璧な日本語文法さえないうえに，人工知能や知識処理の専門家が必死になって構文解析(文の構造を解析すること)を研究している段階である。そう簡単に高度な文法解析能力なんて持たせられるわけがない。だから私はAI辞書だとか意味解析をするなどというワープロは一度疑ってみることにしているのだ。

ワープロだって使い方しだい

Asano Keizou **浅野 恵造**

最近のポータブルワープロやパソコンのワープロソフトが目指す方向として，ひとつにはイメージ取り込みやデスクトップパブリッシングといった，出力結果重視の方向があります。それと同時に，文章を作るという営みに重点を置き，エディタとしての側面を前面に打ち出したものも増えてきています。そして，この後者の方向として，思考を助けるツールとしてのワープロも登場してきています。それが，アウトラインプロセッサなどと呼ばれているものです。

アウトラインプロセッサ

アウトラインプロセッサ，あるいは，アイデアプロセッサというのは，MacのThink TankやMoreといったソフトに代表されるものです。国内では，98用にダイナウェアからプランupというソフトが発売されています。

こういったソフトは，たとえば文章を書く際に章や節などを考えながら全体を構成したりするのに便利にできています。長い文章を書くときに，まず思い付いたことを箇条書きに打ち込みます（メモします）。打ち込みを終えたら（メモしたら）全体を見渡して，順序を入れ換えたり，分類したり，階層化したりします。こういった作業を紙と鉛筆で行うと，何度も書き直したりしなくてはいけませんが，アウトラインプロセッサでは打ち込んだ箇条書きを分類したり，見出しを付けたりすることが非常に容易にできます。そして，誰でも長い文章をまとめ上げることが可能です。もちろん，構成自体の優劣や文体の巧拙には個人差がありますが。

たとえば，Mac用のMoreではマウスを使用して章や節の編集ができるということのほかにも，それぞれの見出し（これをヘッドラインと呼んでいる）の下に文章や図形を入れることができます。また，ヘッドラインを樹系図として表現したり，さまざまなフォーマットでデータベースとして使用したりすることもできるなど，非常に強力な思考ツールとなっています。

思考ツールとしてワープロを使う

ところで，読者の皆さんはワープロをどのように使っているのでしょうか。最近ではかなりの方がワープロソフトを持っていることでしょうし，あるいはポータブルワープロを使っている人もいるかもしれません。また，いずれも持っていなくとも，ワープロには興味を持っているという方は多いのではないでしょうか。そこで，いったいどのように利用しているかを考えてみてください。

多くの方は，挨拶状や手紙，年賀状，暑中見舞いといったもの，またほかにはパーティーなどの案内状，カセットレーベルなどを作成しているのではないでしょうか。そして，少数の人たちが，原稿やレポートの作成といったことに利用しているのでしょう。最初に述べたように，多くの方の利用法というのは，ワープロで綺麗に格好よく印刷できること，すなわち，結果に意味のあるものです。もちろん，私自身もカセットレーベルや案内状の作成などに利用しています。しかし，ワープロの用途はそれだけではなく，編集機能や文章の保存機能を利用して，文章を作成する（思考をまとめ上げる）という行為そのものをバックアップするという捉え方もできると思います。それが思考ツールとしてワープロを利用するということです。

さて，思考ツールといっても実際に前述のアウトラインプロセッサのようなソフトがあるわけではありません。私が使用しているのは，X1turbo用の「即戦力」です。このソフトは，ご存じの方，実際に愛用されている方もいらっしゃるでしょう。いまとなっては派手さはありませんが，変換速度や編集機能など基本的なものをきちんと押さえたワープロです。変換は最近流行の一括変換や自動変換ではなく，熟語単位の変換です。ただし，読みの先行入力は可能で，入力された読みの先頭から最も長く一致する熟語を選択し変換します。再変換や重変換もでき，なかなか効率のよい変換ができます。もちろん，これ以外のワープロソフトやポータブルワープロでもこれから述べるような使い方は可能です。

ところで，アイデアやプランをまとめるときには，人それぞれいろいろなやり方があると思います。私自身もその場その場で違ったやり方をしますが，一般的に心掛けているのは，思い付いたことはできるだけメモしてしまうということです。人間というのはやはり物事をよく忘れるようで，私も自分の記憶力を当てにせず，とりあえずメモするようにしています。メモすることで，あとになって忘れてしまうということもなく，また，一度習慣になってしまうと安心してアイデアを考えることができるようになります。その場合にルーズリーフ式のノートを使いますが，これは，あとで順番を入れ換えたり，分類して綴じることができるからです。こうして考えてくると，メモはすべてワープロに行ってしまうのがいちばんよいということになります。ただし現状では，パソコンにしてもポータブル

ワープロにしても大きさが問題です。しかし，とにかくパソコンに向かってアイデアを練るときには，ワープロを活用することができます。

先ほど，「即戦力」を使用しているといいましたが，このワープロで普段構想を練るときのやり方を簡単に披露しましょう。構想を練るといっても，基本的にはひとつの文章を作り上げることになります。ただし，レポートというかたちで人に発表することを前提としているわけではありません。あえていうならば，未来の自分宛のレポートということになります。

まず，構想なり文章のタイトルを決めます。そして，実際の内容を書き始めるわけですが，そのときに思い付いたアイデアやフレーズは，順番や位置にこだわらずとにかく入力してしまいます。ある段落を入力しているときに，それ以降の文章のアイデアが浮かぶというのはよくあることです。あとでそのことを書こうと思っていても，途中の入力をしていると結構忘れてしまうものです。そこで，私の場合は，思い付いた項目を箇条書きで文章のいちばん最後に入力してしまいます（「即戦力」では，[SHIFT]+[ROLL UP]で文末にカーソルが移動します）。そして，入力していた箇所でひと区切りついた段階で文末のメモを見直します。そのときに，メモの順番を入れ換えたり，字下げをしてメモ同士を階層化して上下関係をはっきりさせたり，あるいは，メモの内容の補足をしたりします。

ところで，「即戦力」には移動や複写の機能がありますが，箇条書きなどは1行以内の場合が多く，いちいち移動や複写を選択するのが面倒です。そこで，行削除と行復活を組み合わせて1行単位の移動や複写などを行うことができます。この場合の行削除や行復活は，ファンクションキーを利用するのではなく，コントロールキーとの組み合わせで行います。つまり，行削除は[CTRL]+[G]で行い，行復活は[CTRL]+[P]を使ってします。たとえば，1行単位の移動は移動したい行にカーソルを持っていきそこで［CTRL]+[G]を押します。するとその行が削除されて，内容がバッファに保持されます。

次に，移動したい先へカーソルを動かしその位置で［CTRL]+[P］を使って内容をコピーします。また，1行単位の複写であればいまの方法で最初に1行削除したあと，すぐその位置で［CTRL]+[P］を実行します。こうすると削除された内容が削除された位置にもそのまま複写されますので，結局，1行複写を実行したのと同じ結果になります。これらの方法は，1行単位であればより簡単に行えるというメリットと，移動先や複写先に同じ内容を何回でもコピーできるという便利さがあります。

この1行移動・複写などの機能を利用して，箇条書きに加筆修正をしていきます。アウトラインプロセッサでは，このような箇条書きの構成を行いやすくしてあります。しかし，そうでないワープロでも箇条書きをあとから階層化したり，順番を入れ換えたりしながら構想をまとめることができるのです。

以上のような使い方をすると，アイデアやプランのまとめがやりやすくなります。また，長い文章を書くのが苦手な人でも比較的容易に文章が書けるようになります。思い付いたことをその場で入力する習慣を付けてしまうと，あとでその入力（メモ書き）を見直して，そこから再び発想の刺激を受けることもあります。このように，入力の便利さや綺麗な印字結果を求めるのとは違う方向のワープロの活用もあるのです。私自身は，知的ツールとしてのパソコンの利用法として今後より一層こうした方向でのソフトの充実を望むと同時に，ユーザーの皆さんも一歩踏み込んだ使い方を工夫してほしいと思います。ハガキなどの印字に飽きてしまった方も，もう一度ワープロの別の可能性を探ってみてはいかがでしょう。

それでも私はWordStar

Katsumoto Shin 勝本 信

初めての出会いから奥義まで

WordStarと初めて会ったのは5年前，N5200のCP/M-86上でのことだ。カーソルの移動は矢印キーではなく，コントロールキーとほかのキーを同時に押すことによって行われる。画面の端までカーソルが行くとスクロールされずに画面全体が書き直された。どんなマシン上でも動くソフトウェアとはこんなふうにして作られているのかと，そのノウハウに感動した。

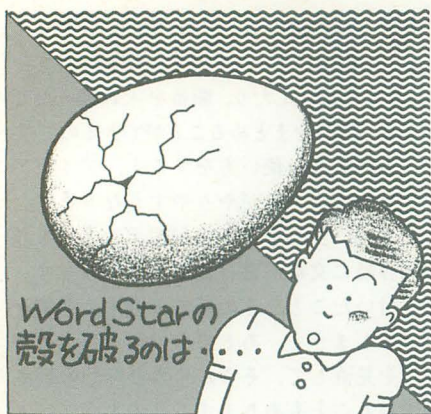
それから5年たった。WordStarで論文も書いた。プログラムも作った。OSもハードウェアも進化した。OSはエスケープシーケンスで画面スクロールができるようになり，WordStarの速さが倍になった。キーボードのキーは自由に再割り当てができるようになり，矢印キーにコントロールコードを割り当てた。便利になった。

インストールプログラムでWordStarを書き換えた。メニュー表示の色を変え，遅延ルーチンの時間待ちを短くしてスピードを上げた。画面表示を40行にしてくれるAdvanced-bitsと組み合わせて大画面を実現した。

WordStarには奥義がある。オペレーションに奥義があり，果てしないテクニックがある。コントロールキーによる2文字コマンドの組み合わせに不可能はない。文書ディスクがいっぱいになったときどうするか。ファイル全体を領域指定して別のディスクに書き込めばよい。操作は^QR^KB^QC^KK^KWでファイル名入力する。

すべての行の先頭にスペースを入れたいときはどうするか。^QAで検索置換を行えばよいが，行の先頭は改行コードすなわち^PM^PJで表し，これを^PM^PJ_に置き換える（_はスペース)。オプションスイッチは全領域にわたって検索置換を実行するG（グローバル）と，確認を行わずに強行するNを付ける。

$\frac{1}{2}$などという分数を印字したいときはどうすればよいか。上付き下付き印字と重ね打ちを併用する。正解は^PT1^PH－^PH^PV2とする。

WordStarの文書ファイル形式も調べた。まだCP/M-86を使っていたころ，DDTでファイルを読み込んで調べた。WordStarの改行には硬い改行と柔らかい改行とがある。スペースにも硬いスペースと柔らかいスペースがある。柔らかいスペースは伸縮自在で，自由に伸び縮みして単語間の間隔を調整し，自動的に右端を揃えてくれる。硬いスペースや改行はその名のとおり堅固で，^Bによるリフォームでも変化しない。この柔らかいスペースと改行のおかげで，1行の長さを自由に変えることができる。

柔らかいスペースと改行の最上位ビットは必ず1となる。だからWordStarの文書ファイルをTYPEコマンドで表示させると漢字やグラフィックキャラクタが出てくる。通常のASCIIファイルに変換するには，PIPコマンドのZオプションなどで最上位ビットをマスクしてしまえばよい。その代わり^Bでリフォームすることはできなくなる。ところでこの便利なZオプションはMS-DOSには存在しない。また，逆に通常のASCIIファイルをWordStarのファイルに直すには，スペースと改行のみをサーチして最上位ビットを立てるプログラムを作ればよい。

WordStarと互換性のあるエディタを持つTURBO PASCALも長い間使ってきた。PASCAL言語も修得した。PASCALを使うためにTURBO PASCALを導入したのではない。TURBO PASCALを使うためにPASCALを学んだのだ。

WordStarが奥義を究めるべき拳法ならばTURBO PASCALはもはや信仰に値する宗教である。実行速度が少し遅い？　時間のかかるルーチンをマシン語でインライン展開しなさい。64Kバイト以上のデータ（MS-DOS版）が扱えない？　動的変数をスタック上に確保しなさい，そうすれば実装メモリぎりぎりまでの配列を使用できます。エディタで漢字がうまく使えない？　日経MixかPC-VANをアクセスしてパッチ情報をダウンロードしなさい。

WordStarに不可能はない

WordStarで日本語のファイルを扱うはめになったこともあった。ファイルの大きさが100Kバイト以上であり，手持ちのどのワープロソフトでも扱うことができなかったからである（X68000付属日本語ワープロ，MZ-6500ダイナデスク，MZ-2800書院28）。

WordStarは日本語ファイルを読み込むことは読み込むのだが漢字の表示はもちろん行われず，わけのわからない文字列が並ぶだけである。しかしセーブは正常に行われることを確かめてあった。そこで，まるで暗闇のなかを手探りで進むようにファイルのなかほどまで行き，前半部分を領域指定してセーブして削除。その後，残った後半をセーブ。ファイルは無事に分割され，ことなきを得た。

OSを変えさせたWordStar

繰り返すがWordStarに不可能はない。MS-DOSの階層化ディレクトリをサポートしていないという非難はあった。これは明らかにMicroPro社の手抜きである。ところが，折れたのはMS-DOSを発売しているMicrosoftのほうだった。ディレクトリを仮想的なドライブ名に割り当てるSUBSTコマンドがMS-DOS ver.3でサポートされたのだ。米国におけるWordStarのユーザーの力がいかに強いかを感じさせる。いやそれよりも，アプリケーションのユーザーがOSそのものを変えさせたことに注目したい。

WordStarを超えるのはWordStar ver.4のみである。今春，劇的にNewStar社の買収を行ったMicroPro社は，NewWordのノウハウを存分に吸収したWordStar ver.4をリリースしようとしている。メーカーの発表によれば百数十項目に及ぶ改善がなされているが，なかでも最大のポイントは，階層化ディレクトリのサポートと，カラム編集モード（四角形の領域指定などが行える）であろう。そのほかにも画面上で下線と太字の表示を行う機能や，22万語に及ぶ類義語辞書，プロポーショナルスペーシング印字，アンドゥ機能，関数電卓機能などが追加されているという。

いま手元にあるのはマウス。目の前にはマルチウィンドウ。一見便利なようだが，感性を刺激する響きが指先に伝わってこない。実際の仕事をこなすにはあまりに機能不足で役に立たない。ユーザーフレンドリという甘い言葉の影に，なにかが忘れられている。

このまま古い殻にとじこもっていたのでは新しい世界は開けてこない。このむなしさはいったい誰のせいなのか。WordStarフリークを満足させる新しい殻（シェル）はいつになったら現れるのか。そんなことを考えながら，今日もゆっくりとWordStarのアイコンをダブルクリックすることになる。

ソフト漫評家は語る

Tachibana Kaoru **立花かおる**

8本あった

私は市販ソフトの収集を趣味のひとつにしている。そのうちワープロがどれくらいあるかを探してみたのだが，案の定，結構出てきた。ユーカラK2，しのぶれど，一太郎，SUPER春望，プランup，クイーン，テラ3世，大名。このほか日本語フロントプロセッサのVJE-α,β,Σ, ATOK5，松茸，刀も見つかった。探せばほかにもまだありそうだ。

もちろん，これらすべてを自分で購入したわけではなく，サンプル版をいただいたり，友人から「正当なサブマスター」注1)を回してもらったりしたものもある。いずれにせよ，沢山あることに変わりはない。

これでもまだまだ不満足だ。いま発注しているのがThe WORD, VJE-PEN，デスクupといったところで，ほかにもEgWORD, Z'sWORD，ユーカラartなど欲しいソフトはいっぱいある。

なぜこんなに持っているのだろうか？　1本使えるソフトがあればいいはずなのに。この原稿を執筆しながら，ふとこのように自問してみた。理由はどうやら，新発売のインスタントラーメンを必ず食べてみたくなる衝動とさして変わらないだろう。

まず，一度使ってみたい。

これがすべてのスタートになっているようだ。そして使ったことがない人に会うと自慢をする。そう，小学校の教室で友人によく話した「ねーねー，あれ，知ってる？」というやつだ。あえてもうひとつ理由をこじつけるとすれば，「使えるソフト1本」を探して歩く求道者の気分に似た側面を自分の中に見いだせることだろうか。特に私のような原稿を書く機会が多い人間にとっては，ワープロは「筆記用具」であるからほかのグラフィックツールやデータベースよりもこだわりが強い。

好き嫌いを考える

当然のことながら，所有しているソフトをすべて使い込んでいるわけではない。比

較的よく使っているのは，MZ-2500用ユーカラK2[注2)]，PC-9801用一太郎，プランup，PC-98LT用大名の4本だけ。そのほかはたまに使えばいいほうで，ほとんど死蔵に近い。

また当然，好き嫌いがある。立場を明確にするために，あえていくつか感想を列記しよう。

▷一太郎＝使わないと世の中から取り残された気分になる脅迫観念がある。

▷ユーカラK2＝スクロールの遅さは表彰ものだが，なんといっても変換が賢い。

▷大名＝プルダウンメニューは満足がいく[注3)]。しかし印刷用には使いたくない。最低だ。

▷テラ3世＝大っ嫌い。

▷しのぶれど＝評価対象外。

▷SUPER春望＝………（絶句）。

このなかには機能が低くて嫌っているものもあるのだが，それ以前に好き嫌いの理由があることに気づいた。ATOK，VJE，ユーカラ，松茸などの系統はスペースバーかXFERキーで変換，選択し，リターンキーで決定する。しかしFIXER系のテラやクイーン，しのぶれどはスペースキーで決定する。春望もそうだ。つまり操作が逆なのである。このため誤入力が跡を絶たず，文書作成時間が大幅に狂ってしまう。

もちろんFIXER系のファンもいるだろう。どちらが正しいとはいえない。しかしこのあたりの初歩的な操作方法は統一してもらったほうが絶対に使いやすいし，それが望まれているはずだ。やはり一太郎，ユーカラ，松を抱えるスペース&XFER変換勢がメジャーであるのは否定できないところ，というのはあくまでも私の個人的な見解に過ぎないのだが……。

マニュアルにもひとこと

マニュアルの読みやすさも課題だ。すべて読みづらい，と私は一刀両断のもとに切り捨ててしまう。

たいていのマニュアルは，まず機能名があってその説明をする，という形式だ。しかし，利用者は「―したいときは？」という形で疑問を持ち，それを知りたいのである。決して「挿入」とか「ブロック編集」の用語説明を知りたいわけではない。全体的に編集方針を変更してもらう必要がありそうだ。

サイズについても問題がある。A4判とか横長型の形はおかしい。片手でパラパラとめくれる小さなもののほうが読みやすいはずだ。コンサイス英和辞典のように，通学・通勤のときにも読めて，カバンにも入り，本棚にも置ける新書判か文庫判のマニュアルをぜひ希望する。

強敵がしのび寄る

シャープからMZ-2861が発売されたが，このマシンの特徴は決して一部の98用ソフトが動くことではなく，シャープ製専用ワープロ書院とコンパチであることだ。というよりもこれ自体が書院であり，パソコン/ワープロ兼用機になっている。そういえば，富士通の新型パソコンFM Rシリーズも専用ワープロOASYSの機能を装備している。そしてなんと，5月のビジネスショウでは，あの[注4)]アスキーがPC-9801向けにOASYSキーボードを参考出品していた。ワープロソフト付きで秋までに発売するという。

パソコンとワープロ専用機の境界がついになくなったわけだ。これまでは「ワープロソフト」か「専用ワープロ」かどちらか一方の知識があれば「先生」だったのだが，今後は両方の知識が求められる。しかし専用機はソフトを手に入れるように簡単には入手できない！

それはともかく，これまでは「私はパソコンマニアなのだ」といえば専用ワープロには背を向けられた。しかし，事ここに至ってはそれはできない。これまではパソコンは多少使いにくくても「パソコンだから」とすませてこられた。しかしそういっている間にワープロはどんどん使い勝手がよくなってきた。このあたりで改めてパソコンに使いやすさを求めるべき時期がきているのかもしれない。

パソコンはもう，われわれマニアだけのものではないのだから。

注1）私の場合，特殊なコピーツールを使う不法コピーは断じてしない。しかしサブマスターをくれるなら話は別だ。

注2）東海クリエイトさんへ：MZ-2500用ユーカラartはどうして発売しないのですか？

注3）大名は「Shogun」の母体になっている。操作性は酷似。

注4）特に意味はない。

ホントに私的なワープロの話

2年ほど前，Oh! MZでワープロ特集を組んだとき，編集室に読者の方から何本か電話がかかってきたそうです。その電話のいずれもが「ワープロを購入したいと思っているんですけど，どのワープロが使いやすいでしょうか」といった内容のものだったようです。

あのころは，ちょうどワープロの購入を考えていた方が多かったせいかもしれませんが，私でもそのような質問をいただいたとしたら「ご自分で使いやすいと思ったものをお選びください」とご返事差し上げるしか方法はないのです。それは今回の特集をお読みになっておわかりのように，祝一平氏は彼なりの，清水和人氏はまた自分なりのワープロを欲しているからなのです。

その昔，ワープロを使い始めた最初のころは打ち出した文章が自分の手書き文字と違って読みやすく，比較的客観的に読むことができてチェックしやすいなどと思っていたものでしたが，ワープロを使うことに慣れてしまうと，結局は同じになってしまうようです。それでも原稿用紙のマス目をにらんで「ああでもない，こうでもない」と思案しながら鉛筆と消しゴムを使っていたときと感覚的にはほど遠いようです。これは私のまったく個人的な話になってしまいますが，その原稿用紙のマス目と格闘していたころは，余白にメモ書きや落書きをしたり，またジッと白いスペースを見ていると突然，原稿用紙の上に絵や写真がボンヤリと浮かんできたりしたものです。そんなことを繰り返しているうちに，突然筆が走り出すといったようなことがずいぶんあったような気がします。

原稿用紙のサイズなんてディスプレイのサイズとそう変わらないはずなのに，ワープロを使っているときには，そんな空想のできる空間が私には感じられないのです。

ワープロを仕事で使っている分には，さまざまな編集機能が付いているおかげで，原稿用紙を使っていたときよりも確かに効率的に仕事が行えるようになっています。私の現状の仕事レベルでは，いま発売されている400～500万円クラスの専用機（UA-100など）があれば，自分がワープロ上で処理したいと考えていることのほとんどのことができてしまうような気もします。しかしそれはあくまでも入力，編集，出力，データ処理といった実務優先の希望的観測であって，実際に私的文書を作成するにあたっては，きっととんでもない発想のワープロを要求することになるのでしょう。

そういった意味からも，私の本当に使いたいと思うワープロとは，きっと自分の部屋に掛けてある1枚の絵と同じようなものなのでしょう。毎日見ていても飽きない，その日の気分によって違って見える，そしてその場所にいれば落ち着いてものごとを考えられる。これは少し贅沢すぎたでしょうか。ただディスプレイの向こう側に自分だけの空間がある，そんな感情が抱けるワープロが1台あってほしいものです。（T.S.）

書式ユーティリティCOLN

MZ-2500，X1turbo用

Izumi Daisuke
泉 大介

MZ-2500V2のアルゴエディタはちょっとの工夫で市販の日本語ワープロ以上の威力を発揮する。こうして泉先生は今日もOh! MZの原稿を書いていたのであった。

このプログラムは1985年6月号のOh!MZで発表された中川智哉氏の書式設定ユーティリティを下敷きに，MZ-2500のアルゴエディタで文書を書く際に便利なよう改造を加えたものです。中川智哉氏のものはX1 turbo用で，BASICのAUTO＊によって書かれた文書を決まった字詰めでプリンタに打ち出すユーティリティでしたが，今回発表するものは決まった字詰めの同名のファイルを作るというものです。MZ-2500版とともにX1turbo版も用意しましたので活用してください。

速くなければスクロールではない

ずっと原稿用紙に原稿を書いてきた私は，文書の保存という目的もあってMZ-2500で原稿を書くことを決意します。そして市販品の中で使い物になるのはまァこれしかないだろうという編集者の勧めから某ワープロを使い始めました。こいつはプリントアウトする文書のイメージどおりに画面に表示してくれるため最初は珍しさも手伝って喜んで使っていたのですが，しばらく使っているうちにうっとうしくてたまらなくなってきました。

私の場合，書いている最中は今までに書いてきたところを何度も読み返して推敲していきますので，当然数ページ前とか後とかを頻繁に参照します。ところがこのワープロは，1ページ上を見ようとページスクロールさせると，画面を「ちん…たら…たら…たら……」とまるであくびが出そうな速度で書き直すのです。もっとひどいのは画面のいちばん上でカーソルを上に動かしたとき，すなわち画面を1行スクロールさせたときです。1画面全部を同じ調子で書き直すため，待たされるほうは地獄を見ます。印字イメージを画面上で確認できるようにするため，文字をグラフィック画面に書いているのです。いかにZ80Bといえどもこんなことをされてはたまったものではありません。これではせっかくの漢字VRAMが泣きます。

ついに怒りが頂点に達しようとしたある日のこと，MZ-2500用にV2 BASICが発売されました。この新しいBASICにはおまけながらもスクリーンエディタが付属しており，しかもアルゴキー一発で呼び出して使うことができるようになっていました。もちろん漢字VRAMを使っていますので，そのスクロールは某ワープロとは比べものにならない速さです。私が速攻でアルゴエディタに乗り換えたのはいうまでもないことです。

アルゴエディタではかな漢字変換は内蔵の辞書ROMによって行われます。この辞書ROMはなかなかにバカなのですが，ROMであるため変換が高速であるという唯一の利点を持っています。またV2になって学習機能も付きましたので，そこそこ便利に使うことができるのです。ただしさすがにオマケだけあってアルゴエディタには文書の横幅を設定する機能などは付いていませんでした。

この原稿は19字詰めで書いていますし，マシン語体操の原稿は29字詰めで書きます。最初は手作業で直してみました。しかしこれでは揃えたあとで再び推敲して数文字書き加えた場合，削った場合に地獄です。やっぱりあのワープロを使うしかないのだろうか。非常に暗い気持ちでそう考えていたとき，以前turboの日本語BASICがワープロ代わりに使えるという話が載ったことを思い出しました。

文書も美しくありたい

さっそく中川智哉氏にいってX1turboからMZ-2500にRS-232Cでプログラムを転送してもらいました。この書式設定ユーティリティは書式設定してプリンタに打ち出すだけのものなので，私の要求にはちょっとかないません。しかし，プリンタに打ち出す部分を変更してディスク上のファイルに書き出すようにすれば，いままでデコボコして見づらかった文書をきちんと整形した文書に変えてやることができます。

そうしてできあがったのがリスト1のプログラムCOLNです。このプログラムでは整形した文書をいったんTEXTというファイル名で作成し，作業が終わったあとで元のファイルをKILLして，TEXTをRENAMEしています。そのため作業するディスク上にTEXTというファイルが存在していてはいけません。注意してください。

プログラムは120〜140行にパラメータを設定して使います。まず120行には整形しようとする文書のファイル名を，130行には1行に何文字の全角文字を入れるか，最後の140行は禁則処理をするかどうかのフラグです。140行でKSを1にすると禁則処理を行うようになっています。

このCOLNでは“▼”を改行マークとして用いており，通常の改行は無視されます。“▼”が気に入らない方はリスト中の該当部分をお好みの全角コードに変えてください。

●MZ-2500の場合

まず最初にCOLNをロードし，必要なパラメータを全部書いておいてください。

MZ-2500ではアルゴエディタを使って文書を作成します。アルゴエディタを立ち上げたら，最初に“▼”を入力しリターンキーを押します。次にファンクションキーの7番の「行コピー」を使って，“▼”だけの行を20行くらい作ります。先ほども説明したように，改行は“▼”のところで行われます。通常の改行マークは無視して文書の桁数を揃えますので注意してください。

文書ができたらCOLNに設定したのと同じファイル名で文書をセーブし，アルゴエディタを抜けます。現在メモリ上にはCOLNがありますからファンクションキーの5番で実行すればいいだけです。整形が終わると「ピッ」という音が鳴って終了を知らせます。再びアルゴエディタを立ち上げて作成した文書を読み込んでみてください。綺麗に整形されているでしょう。

アルゴエディタをお持ちでない方は，次のX1 turboと同じ要領で作業を行ってください。X1turbo用の3000行以降を加えたものをCOLNとして使います。

●X1 turboの場合

まずBASICのAUTO＊を使って文書を作ってください。改行マークの“▼”はファンクションキーに登録しておき，改行したいところでファンクションキーを押すようにすれば便利でしょう。文書が作成できたら「LIST＊“ファイル名”」命令で文書をセーブします。

次にCOLNをロードし，必要なパラメータを設定してRUNしてください。整形された文書が作成されます。ただしこれは文書の整形しか行いませんので，このままでは再びエディットすることができません。そこで再びエディットを続けるなら「GOTO 3000」としてください。3000行からは文書をBASICのプログラムに直すユーティリティが入っています。くれぐれも「RUN 3000」としないようご注意ください。

「GOTO 3000」の実行が終わったら文書ファイルをロードしてみてください。整形された文書が見られます。再びAUTO*で続きを入力していってください。

COLNのおまけ

COLNには2つのおまけが付いています。ひとつは1000行からのプリントアウトプログラム。もうひとつは2000行からの行数カウントプログラムです。

プリントアウトプログラムは文書中の“▼”を取ってプリンタに印字するためのプログラムです。COLNをRUNした直後なら（MZ-2500ならCOLNを一度でも実行したあとならいつでも）「GOTO 1000」とやるだけで文書をプリンタに出力します。新たにCOLNをロードしたならFILNAM$という変数にプリントアウトしたいファイル名をダイレクトモードでセットし「GOTO 1000」とやって使います。

130桁のプリンタをお使いの方は1040行の「FOR I=0 TO 19」を「FOR I=0 TO 39」に変えれば2段組で文書を出力することが可能です。現在の値は1行19文字で打ち出したときに綺麗に表示されるようになっていますので，打ち出す文書の1行の文字数に合わせて1150，1170行を適当に変更して使ってください。

次の行数カウントプログラムはアルゴエディタがいま何行目なのかを表示してくれないために付けた機能です。プリントアウトツールと同じ要領で使います。

＊　＊　＊

MZ-2500ではアルゴエディタとこのCOLNのコンビは最強であると自負しています。文書が長くなりすぎると整形に時間を取られますが，それでもスクロールスピードは魅力です。

あとはMZ-2500の辞書がなんとかなればなぁと思いつつアルゴエディタを抜けることにします。勝手気ままに書いた文書を整形してくれるCOLN。皆さんもぜひ使ってみてください。

リスト1　書式ユーティリティCOLN

MZ-2500(BASIC-M25 V2.0)用

```
100 REM <<<<< WP-Print Utility >>>>>
110 CLS:DEF INT A-Z
120 FILNAM$="マシン語 20"                      ' ファイル名
130 WD=29                                      ' 1行の文字数
140 KS=1                                       ' 禁則処理 (1/0)
150 REM <<< read sentence >>>
160 CLS:N$=""
170 OPEN "I",#1,FILNAM$ : OPEN "O",#2,"TEXT"
180 WHILE EOF(1)=0:LINE INPUT #1,M$:GOSUB 220:WEND
190 CLOSE #1:P$=N$:GOSUB 590 : CLOSE #2
200 KILL FILNAM$:NAME "TEXT" as FILNAM$
210 BEEP:END
220 REM <<< print main >>>
230 IF M$="▼" THEN
240   IF N$="" THEN
250     P$=M$:GOSUB 590
260   ELSE
270     P$=N$:GOSUB 590:P$=M$:GOSUB 590:N$=""
280   END IF
290   RETURN
300 END IF
310 N$=N$+M$
320 WHILE LEN(N$)>WD*2+2
330   Y=INSTR(N$,"▼")
340   IF Y AND Y=<WD*2+1 THEN GOSUB 360 ELSE GOSUB 390
350 WEND:RETURN
360 REM <<< return mark >>>
370 P$=LEFT$(N$,Y+1):GOSUB 590:N$=MID$(N$,Y+2)
380 RETURN
390 REM <<< normal print >>>
400 C=WD-1:REPEAT:X=Y:Y=Z:Z=KPOS(N$,C):C=C+1:UNTIL Z>WD*2+1
410 IF KS GOSUB 440
420 P$=LEFT$(N$,Y-1):GOSUB 590:N$=MID$(N$,Y)
430 RETURN
440 REM <<< kinsoku >>>
450 C$=MID$(N$,X,Y-X):ON Y-X GOSUB 530,540
460 IF C THEN IF Y=<W*2 THEN Y=Z:RETURN ELSE Y=X:RETURN
470 C$=MID$(N$,Y,Z-Y):ON Z-Y GOSUB 560,570
480 IF C THEN
490   Y=Z
500   IF MID$(N$,Y,2)="▼" THEN Y=Y+2
510 END IF
520 RETURN
530 C=INSTR("([{｢",C$):RETURN
540.C=INSTR(HEXCHR$("816581678169816B816D816F81718173817581778179"),C$):RETURN
550 C=INSTR("‘“（〔［｛〈《「『【",C$):RETURN
560 C=INSTR(",)]}｣",C$):RETURN
570 C=INSTR(HEXCHR$("8141814281438l4A814B81668168816A816C816E8170817281748176817
8817A"),C$):RETURN
580 C=INSTR("、。，′・’”）〕］｝〉》」』】",C$):RETURN
590 REM <<< print out >>>
600 PRINT #2,P$
610 RETURN
1000 '---------------------
1010 OPEN "I",#1,FILNAM$
1020 DIM LIN$(39)
1030 LPRINT:LPRINT
1040 FOR I=0 TO 19
1050   IF EOF(1)=0 THEN
1060     LINE INPUT #1,LIN$(I)
1070     IF RIGHT$(LIN$(I),2)="▼" THEN
1080       LIN$(I)=LEFT$(LIN$(I),LEN(LIN$(I))-2)
1090     END IF
1100   ELSE
1110     LIN$(I)=""
1120   END IF
1130 NEXT
1140 FOR I=0 TO 19
1150   LPRINT "      ";LIN$(I);
1160   LPOUT CHR$(13)
1170   LPRINT STRING$(73," ");LIN$(I+20)
1180   LPRINT:LPRINT
1190 NEXT
1200 IF EOF(1) THEN CLOSE:LPRINT CHR$(12);:END
1210 LPRINT CHR$(12):LPRINT:LPRINT
1220 GOTO 1040
2000 '---------------------
2010 CLS
2020 OPEN "I",#1,FILNAM$
2030 I=0
2040 WHILE EOF(1)=0
2050    LINE INPUT #1,A$
2060     I=I+1
2070    LOCATE 0,0:PRINT I
2080 WEND
2090 CLOSE
```

X1 turbo(CZ-8FB02)用追加・変更点

```
110 CLS:DEFINT A-Z
230 IF M$="▼" THEN 240 ELSE 310
240   IF N$="" THEN 250 ELSE 270
250     P$=M$:GOSUB 590 : GOTO 290
270     P$=N$:GOSUB 590:P$=M$:GOSUB 590:N$="" : GOTO 290
480 IF C THEN 490 ELSE 520
500   IF MID$(N$,Y,2)="▼" THEN Y=Y+2 : GOTO 520
510 'END IF
1050   IF EOF(1)=0 THEN 1060 ELSE 1110
1070     IF RIGHT$(LIN$(I),2)="▼" THEN 1080 ELSE 1130
1080       LIN$(I)=LEFT$(LIN$(I),LEN(LIN$(I))-2) : GOTO 1130
1110     LIN$(I)="" : GOTO 1130
3000 '-----------------
3010 OPEN "I",#1,FILNAM$
3020 LNUM=10
3030 OPEN "O",#2,"TEXT"
3040 WHILE NOT EOF(1)
3050   LINPUT #1,A$
3060   A$=MID$(STR$(LNUM),2)+" '"+A$
3070   LNUM=LNUM+10
3080   PRINT #2,A$
3090 WEND
3100 CLOSE
3110 KILL FILNAM$ : NAME "TEXT" AS FILNAM$
3120 END
```

日本語

特集 迷宮の日本語処理環境

らくらくSYMBOL

MZ-2500，X1turbo用

SYMBOL文を使えば，日本語もりっぱなグラフィックである。だが，メモリを食いつくすほど図体のでっかいグラフィックツールよりは，BASIC上のほうがメッセージの処理は簡単なようだ。

Takiyama Takashi
瀧山 孝

先日，吉田幸一氏似のK君がturboZ似の満貫1号上でZ'sSTAFF似のグラフィックツールを使って角のたばこ屋のポスターを作っているのに出くわした。ちょうどイメージスキャナで取り込んだ絵にコピーを入れている最中だった。望みの文字を選ぶのに苦労はしたものの順調に文字が並べられ，あと1文字を残すのみとなったときに彼は大きく溜め息をついた。文字と文字の間隔が狭くなりすぎてしまったのに気づいたのだ。私はマウスをちょちょいと動かせば取り消しがきくものと思って見ていたのだが，そうではなかった。彼は元絵をディスクからロードし，作業を初めからやり直さなければならなかったのである。

君ならどうする

グラフィックツールで文字を使おうと思った場合の問題は2つある。ひとつは文字を選ぶ手順の繁雑さだ。BASICでなら辞書変換で即座に望む「単語」が出てくるのに，たとえばZ'sSTAFFでは50音順に並んだ漢字の長い列から「1文字ずつ」探してこなければならない。もうひとつの問題点は訂正がきかないこと。一度画面に書いてしまえばハイそれまで。KGB(カーゲーベー)のごとく失敗は許されない。せっかく書いた文字は「消される」のみだ。

ここで「こんなもんさ」とあきらめてしまうのが普通の人。先月の特集を読んだ良い子はIPLスイッチをちょんと押してBASICを立ちあげることだろう。もちろんG-RAMを消さない細工をして，だ。それからダイレクトモードでSYMBOL文を使って字を書く？　残念ながらそれでは50点しかあげられない。文字を表示する座標をどうやって求める？　失敗したらどうする？　人事部のおじさんは，いつも無能な奴を探しているのだ。

BASICで簡単なプログラムを作ろうと思った君は90点。というわけで，私なりにプログラムにしてみた。このプログラムが掲載されるのを見越していた君に100点をあげよう。

らくらくSYMBOL

リスト1の上がMZ-2500用で，下がX1turbo用の変更点となっている。美しいBASICプログラムにこだわる私にしては珍しく汚いプログラムになってしまったが，この際それはどうでもいい。それぞれのBASICからぽこぽこ打ち込んでくださいな。

チェックが済んだら，RUNする前に100行以下の部分を画面モードに従って変更する。MAX_Xに画面横ドット数，MAX_Yに縦ドット数，MAX_Cに色数を入れるようにすればよい。どういうわけか1を引くことになっているので横ドット数が640のモードならMAX_X=639とする。たいした意味はないが，いわゆる縁起物だから真面目にセットしておいたほうが無難だ。

走らせると順に表示文字色，横倍率，縦倍率，表示モードを聞いてくるから，正直に答えてやってほしい。手を抜いてエラーチェックをしていないので，ここで変な数字を入れるとあとで困るのは君だ。ただし，表示文字色が負の値であればプログラムは実行を終える。表示モードは標準，影付き，袋文字をそれぞれ0,1,2の数字で指定する。モードで標準以外を選んだ場合は，さらに影の色（袋文字では輪郭の色）を聞いてくるから，やはり正直に答えること。さらに影付きにしたい場合は影を右下に付けるか左下に付けるか聞いてくるから1か-1で答えてあげるとプログラムも喜ぶ。

ここまで設定したら，表示する文字列を入力する。入力がヌルストリングだと設定画面に戻る。文字列を入力すると画面の適当な位置に白い枠が現れるだろう。この枠が表示する文字列の大きさだ。このとき，倍率によっては枠が画面をはみ出してしまうことがある。強行するとのちのちエラーが出てプログラムが止まるから（ひでぇ），ESCキーを押してほしい。と，文字列の入力待ちに戻すことができる。枠が画面をはみ出さないときでも，「やっぱ，やーめた」キーとしてESCはいつでも君を待っているから，安心して作業を続けられるだろう。

しかもおおかたの予想どおり，この枠はテンキーであっちこっちと動く。スピードを優先させたため動きは粗い。もしかすると望みの位置にぴったりと止めることができないかもしれない。が，とにかく近くまでは持っていけるから，ここがいちばん近いという場所にきたらポンとリターンキーを押す。すると，いかにもSYMBOL文で書いてますという感じで文字が表示されるはずだ。市販のグラフィックツールはここまでしかやってくれないが，このプログラムはここから本領を発揮する。

まず，この状態でもESCキーは「やっぱ，やーめた」キーとして機能する。さらに，表示した文字はテンキーでズリズリと動かすことができる。この移動は1ドット単位だから，こんどこそ望みの位置へ文字を表示したいという願いがかなえられるだろう。「もう絶対に動かさないぞ」と決めたらリターンキーを押せば文字は固定され，再び文字列の入力待ちとなる。リターンキーを押したあとで「失敗した！」と思っても，このプログラムではもはや修正することはできない。

続いて同じ色，同じモードで文字を表示したければ次の文字列を入力して作業を続ければいいし，モードなどを変えたければ空文字列を入力して設定画面に戻ればよい。すべての作業が終わったら，またIPLスイッチのお世話になってグラフィックツールに戻ることになる。MZ-2500では全画面をセーブしてもよい。

復活の秘密と改造の手引き

このプログラムではしつこいぐらいに前の画面に戻ることにこだわっている。なかでもポイントは文字をSYMBOL文で表示してから，元どおりにする部分だろう。が，なんのことはない，画面の一部を退避させておくだけのことなのだ。turbo版では誰でも思いつくようにGET@文を使っている。2500でも同様の方法を使ってもよかったのだが，より融通がきくようにGSAVEを利用してMEM:に退避させる方法をとった。両者共にあらかじめ確保した配列およびMEM:の大きさが退避できる画面の大きさ(=表示する文字列の長さ,倍率)に制限を与える。2500ならこの制限をとっぱらうことは可能だ。プログラム中のMEM:を1:にで

も変更すれば2DDのディスクをワークに使えるからだ。しかし，turboではこうはいかない。それでも，G-RAMの裏半分が使える状態であればVDIMでかわすこともできるだろう。

遅くてもよければ小さなブロックごとにGET@してディスクに書き出す手もある。EMM:が使えればベストだ。でなければ，画面を退避させるのをやめて，SYMBOL文をXORモードで実行するように変更するのもよい。その場合リターンキーで位置が確定したときにもう一度PSETモードで書いてやる必要がある（この方法がいちばんよかったのかもしれないな，とふと思う私)。

まちがっても，配列の大きさをけちってはいけない。特にturboユーザーがこのプログラムを使うときには，配列をできるだけ大きく確保し，極力大きな文字を表示しないよう心がけてほしい。プログラム側では一切チェックしていない。なお，配列の大きさはプログラムの先頭でDEKIRUDAKE_IPPAIという変数で定義されている。

さて，このプログラムはまだいくらでも機能を付け加えることができる。たとえば，現在は影を左下か右下にしか付けることができないが，上や真横に付けるようにしてもよいだろう。また，縦書きのモードもあれば便利に思う。斜体もサポートしたかったところだ。手を抜いたエラーチェック部分も付け加える価値がある。BASICで書かれたお絵描きツールがあれば，このプログラムを組み込んでしまうのもお得かもしれない。

あと，MZ-2500でカラーシミュレータが使えないことについさっき気がついた。設定画面ではグラフィックを消しているのでカラーシミュレータが表示されないのだ。256色モードのときに色を指定するのにカラーシミュレータがあると楽だから，適当に改善したほうがよいだろう。

こんなところで，日本語処理特集というよりも先月のグラフィック特集から出張してきたような文章は終わりを告げる。

じゃ，ね。

リスト1　らくらくSYMBOL

MZ-2500用

```
 10 '/*
 20 '  らくらくＳＹＭＢＯＬ　ＭＺ-2500用
 30 '            written by たーくん
 40 '                                 */
 50   init "crt:80,25,1,0"
 60   init "mem:0,0":init "mem:"+str$(fre(4)/1024)
 70   gsave "mem:temp",(0,0)-(0,0)
 80   klist 0:kmode 1
 90   CR$=chr$(13):BK$=chr$(27)
100   MAX_X=319 :'画面横ドット数－1  (319/639)
110   MAX_Y=199 :'画面縦ドット数－1  (199/399)
120   MAX_C=15  :'色数－1           (3/15/255)
130   init "crt2:"+str$(MAX_X+1)+","+str$(MAX_Y+1)+","+str$(MAX_C+1)
140 '設定
150   view@ (1,1)-(0,0):cls:locate 0,0,2
160   input "                         表示文字色 ",C
170   if C<0 then print chr$(4,12):end
180   input "                              横倍率 ",MUL_X
190   input "                              縦倍率 ",MUL_Y
200   input "モード .0(標準),1(影付き),2(袋文字) ",MDE
210   if MDE=0 goto 250
220   input "                                影色 ",SHDW_C
230   if MDE=2 goto 250
240   input "           影の向き 1(右側),-1(左側) ",SHDW_D
250   X=0:Y=0
260 'メインループ
270   locate 0,0,2
280   view@ (0,16)-(MAX_X,MAX_Y)
290   cls:input "表示する文字列:",PR$
300   locate 0,0,0:cls:view@ (0,0)-(MAX_X,MAX_Y)
310   if PR$="" then 140
320   SIZ_X=(len(PR$)*8+MDE)*MUL_X-1:if SIZ_X>MAX_X then SIZ_X=MAX_X
330   SIZ_Y=(16+MDE)*MUL_Y-1:if SIZ_Y>MAX_Y then SIZ_Y=MAX_Y
340   repeat
350     line (X,Y)-(X+SIZ_X,Y+SIZ_Y),xor ,,B
360     A$=input$(1):DONE=(A$=CR$)
370     A=val(A$):if (A=5) or (A=0) and (A$<>BK$) and (A$<>CR$) then 360
380     if A then 390 else 430
390       DX=((A+2) mod 3-1)*8:DY=(2-(A+2)¥3)*8
400       line (X,Y)-(X+SIZ_X,Y+SIZ_Y),xor ,,B
410       gosub 740:goto 460
420 '   else
430       line (X,Y)-(X+SIZ_X,Y+SIZ_Y),xor ,,B
440 '   end if
450     if A$=BK$ then 260
460   until DONE
470   repeat
480     kill "mem:temp"
490     gsave "mem:temp",(X,Y)-(X+SIZ_X,Y+SIZ_Y)
500     on MDE+1 gosub 650,610,680
510     A$=input$(1):DONE=(A$=CR$)
520     A=val(A$):if (A=5) or (A=0) and (A$<>BK$) and (A$<>CR$) then 510
530     if A then 540 else 570
540       DX=(A+2) mod 3-1:DY=2-(A+2)¥3
550       gload "mem:temp",(X,Y):gosub 740:goto 590
560 '   else
570       if A$=BK$ then gload "mem:temp",(X,Y):goto 340
580 '   end if
590   until DONE
600   goto 260
610 '影
620   for I=SHDW_D to SHDW_D*MUL_X step SHDW_D
630     symbol (X+I,Y+MUL_Y/MUL_X*abs(I)),PR$,MUL_X,MUL_Y,SHDW_C
640   next
650 '標準
660   symbol (X,Y),PR$,MUL_X,MUL_Y,C
670   return
680 '袋
690   for I=0 to 2:for J=0 to 2
700     symbol (X+MUL_X*I,Y+MUL_Y*J),PR$,MUL_X,MUL_Y,SHDW_C
710   next:next
720   symbol (X+MUL_X,Y+MUL_Y),PR$,MUL_X,MUL_Y,C
730   return
740 '移動
750   X=X+DX:Y=Y+DY
760   if X+SIZ_X>MAX_X then X=MAX_X-SIZ_X
770   if X<0 then X=0
780   if Y+SIZ_Y>MAX_Y then Y=MAX_Y-SIZ_Y
790   if Y<0 then Y=0
800   return
```

X1turbo用追加・変更点

```
20 '  らくらくＳＹＭＢＯＬ　Ｘ１ｔurbo用
50 '
60   DEKIRUDAKE_IPPAI=1000 :'画面退避用配列の大きさ
70   DIM temp#(DEKIRUDAKE_IPPAI)
120 '
130 '
150   PALET@ 0,0,0,0,0,0,0,0:CLS:KEY 0,""
170   IF C<0 THEN INIT:CLS:END
270   KEY 0,"":PR$=""
280   PALET@ 0,1,1,1,1,1,1,1
300   CLS:PALET@ 0,1,2,3,4,5,6,7
340   KEY 0,"":REPEAT
360     GOSUB 810:DONE=(A$=CR$)
470   KEY 0,"":REPEAT
480 '
490     GET@ (X,Y)-(X+SIZ_X,Y+SIZ_Y),TEMP#,7
510     GOSUB 810:DONE=(A$=CR$)
550       PUT@ (X,Y),TEMP#,PSET,7:GOSUB 740:GOTO 590
570       IF A$=BK$ THEN PUT@ (X,Y),TEMP#,PSET,7:GOTO 340
810   REPEAT:a$=INKEY$(0):UNTIL a$<>"":RETURN
```

特集 迷宮の日本語処理環境

Superものかきくん

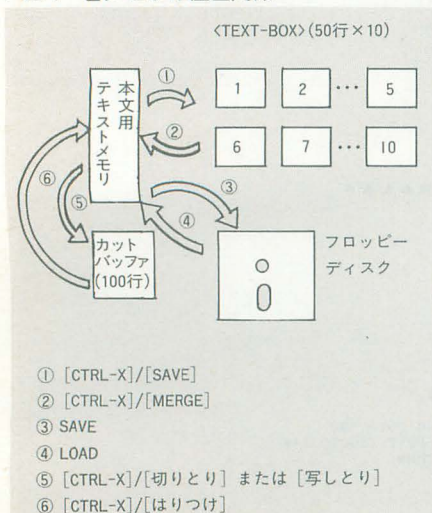
Takahara Hideki
高原 ひでき

多機能化が進む日本語ワープロ。だが，使い手の思想が反映されたものでなければ，どんな機能も生きてこない。Superものかきくんは，高原氏のネットワーカー魂によって生まれた新概念のワープロだ。

パソコンのソフトといえばなんといってもワープロソフトです。ところが市販ソフトを見ると，機能の優劣こそありますが，どれもこれも似たものばかり。結局は表示される編集画面に文字を並べて，印刷してディスクにとっておくことだけしかできません。ワープロの特質は機能競争にあるのではなく，もっと別なところにあるのではないでしょうか？　売っていないのなら，自分で作ってやろう――こうして誕生したのが今回の「Super ものかきくん」です。一般的な文書作成作業に役立つだけでなく，パソコン通信におけるダウンロード（受信記録）ファイルの整理では圧倒的な威力を発揮します。MZ-2500 ユーザーの中には私と同じくネットワーカー（パソコン通信の利用者）の方が多いようですので，ぜひお試しください。

文書作成という思考作業

もうワープロが「清書機」であるという時代は終わりました。ワープロは「文書作成」という人間の思考作業を手助けするサポートツールであるべきなのです。

■図1　各メモリの位置関係

では文書作成の本質とはなんでしょうか。ズバリ，頭の中にある知識と机の上にある資料をほどよく整理整頓して並べ換えながら表現する。さらにその作業を通していろいろなことを考え出すこと――私はこう考えます。そこで今回の「Super ものかきくん」ではこの「整理整頓」をトータルコンセプトとして打ち出しています。

こまぎれの文章を作って保管しておきあとで組み合わせる。逆に長い文章はいったんバラバラに解体して，目次を作って再編集する。文章ができたら別の文章と組み合わせて編集する……。

このように文章の編集作業を積み重ねていくことで，画面上の文章だけではなく，頭の中も整理整頓ができるわけです。その結果，質の高い文章ができるのです。

この整理整頓機能を実現するために2つの機能を重視しました。「カット＆ペースト」と「テキストボックス」です。

「カット＆ペースト」は同一文章内での整

■図2　TEXT-BOX MENUの表示

パソコン通信のファイルリストがそのまま作成できる(ある日の EYE-NET を編集している図)

```
--------+-<< Superものかきくん >>-+---------+---------+---------
  11 PCK001 87/06/20 06:01 L008 Q>カプラについて／秀
闇のネットワーカー・秀です。いま徹夜残業中なのです。
何とあいさつしたらいいのだろうか？　おはようというには今日は暗い。

音響カプラについての質問です。
LTのおともをするのに向いている安くて軽くて電池で使える
音響カプラを探しています。どなたか、紹介してください。
LAC－300を使っていますが、コードがじゃまなので、嫌いです。

  4 BPJ001 87/06/23 22:53 L004 デバッグとマニュアル作りということは、、、
  1 KGK002 87/06/25 23:09 L005 5GのCP／Mについては
  9 NJG001 87/06/20 21:55 L017 ワードバンクLXは意外に重い
  1 WJJ001 87/05/01 19:02 L052 アイドル・・データバンク＜守谷香＞
  6 PJY001 87/06/27 00:00 L024 「コミック雑誌なんかいらない！」7／4（土）

  9 PJY001 87/06/24 04:22 L030 「アリエスの乙女たち」（第12回）7／1（水）


カーソル: 1  85
# どのBOXを選びますか？（[CR]:決定 [ESC]:とりやめ）
```

■図3　ダウンロードファイルは不要部分でいっぱい

```
 [7]        コンプティーク・アイドルデータバンク
 [8]        ミュージックリサーチ・ヒットチャート
 [?]        ヘルプ
 [EXIT]     終了
==>
2
：まかせてチョンマゲ・ニッポン放送
    掲示総数：    14件
==> [コマンド]
GO 51
：夢咲案内人・フジテレビ
    掲示総数：    49件
==> [コマンド]
L10
（受信）：夢咲案内人・フジテレビ
  1 PJY001 87/06/26 23:27 L051 超過酷な”鉄人レース” 6／28（日）生中継
  2 PJY001 87/06/26 02:04 L039 衛星中継による世界初の日本語ニュース放送
  3 PJY001 87/06/26 01:55 L056 「日本ストリート物語」7月上旬放送予定
  4 PJY001 87/06/26 01:55 L065 「日本ストリート物語」（7／5スタート）
  5 PJY001 87/06/27 00:00 L022 「エレンディラ」7／3（金）
  6 PJY001 87/06/27 00:00 L024 「コミック雑誌なんかいらない！」7／4（土）
  7 PJY001 87/06/26 00:00 L039 「刑事の妻の告白」7／3（金）
  8 PJY001 87/06/25 00:00 L030 「クセになりそな女たち」（第13回）7／2（木）
  9 PJY001 87/06/24 04:22 L030 「アリエスの乙女たち」（第12回）7／1（水）
 10 PJY001 87/06/24 04:21 L031 「夏樹静子サスペンス」（最終回）6／29（月）
==> [コマンド]
D9
  9 PJY001 87/06/24 04:22 L030 「アリエスの乙女たち」（第12回）7／1（水）
後8：00～8：54
　嫉妬のあまり水穂薫（南野陽子）を鉄パイプでなぐろうとして階段から落ち、流
産の危機に直面した結城敬子（相楽ハル子）。薫は必死で赤ん坊の無事を祈る敬子
の姿を見て、結城司（松村雄基）に、自分の事は忘れてほしいと頼む。薫のことを
愛しながらも、幼い命を宿した敬子を捨てられない司は、胸がはり裂けそうな思い
だった。
　薫は病室の敬子に付き添った。しかし、やっと気がついた敬子は、薫に感謝しな
```

理，「テキストボックス」は文章単位での整理用です。図1のようになります。具体的に説明してみます。

カット＆ペースト

まずカット＆ペーストから。これはいわゆる「切りはり」作業です。操作方法はあとでまとめますが，文章の一部を対象範囲として指定し，編集中の文章の中で整理整頓ができるようにしました。見ていて楽しめるように美しいプルダウンメニューやリバース表示を多用しています。

用意した機能は，

・空白行挿入（INSERT）
・切りとり（CUT）
・はりつけ（PASTE）
・写しとり（COPY）
・保管（SAVE）
・合成（MERGE）

の6つです。操作性を高めるためにカットバッファとして100行の一時保管領域を用意しました。CUTとCOPYの作業でここに文章の一部を取り出すことができます。

テキストボックス

これを簡単に説明しますと，短い文章をコマ切れにして収納しておける「ひきだし」です。ひとつは最大50行で10個まで用意しましたが，変数の初期設定を変更することによって変更が可能です。画面上では常に「目次」のように表示するTEXT-BOX MENU（図2）形式で表示します。ここには各文章の1行目を表示する仕組みにしてあります。

収納方法はさきほどのカット＆ペーストでプルダウンメニューが表示されたら「SAVE」を選択します。2つ目のプルダウンメニューが現れましたら「TEXT-BOX」を選択します。これによって，指定領域がテキストボックスに収納されるわけです。

さて使用方法ですが，［CTRL］＋［W］または［F10］を押しますと収納した際に表示されたTEXT-BOX MENUが表示されますので，その中から編集したい文章を選択してリターンキーを押します。するとその文章が「裏の編集画面」に転送され，通常どおり編集できます。そこで［ESC］キーを押してメインメニューに戻ると，印刷，セーブなどの機能が使えます。編集が終了して再びテキストボックスに戻すときには呼び出しのときと同じく，［CTRL］＋［W］または［F10］を押します。

リスト1　Superものかきくん

```
100 ' ==================================
110 '    * Ｓｕｐｅｒものかきくん V1.2
120 '                         S.62 1/1
130 '                   by H.Takahara
140 ' ==================================
150 INIT "CRT:,,,0":NEW ON 1
160 KLIST 0
170 CONSOLE 0,24,0,80
180 CLS 2:COLOR 7
190 UP=30:DN=31:RT=28:LT=29
200 PG=0:'          ページ数（０：表／１：裏）
210 X=1:'           Ｘ座標
220 Y=1:'           Ｙ座標
230 YS=1:'          表示開始行
240 YR=1:'          Ｙ座標の行ポインタ
250 XM=80:'         １行文字数（半角時）
260 YM=299:'        文書の行数
270 YE=1:'          文書の最終行
280 VM=50:'         小文書ファイル V$ の行数
290 MK%=&J2170:'    記号の番号
300 REM ------------ Ｆキーの設定
310    KEY (1),CHR$(1)
320    KEY (2),CHR$(22)
330    KEY (3),CHR$(3)
340    KEY (4),CHR$(10)
350    KEY (5),CHR$(19)
360    KEY (6),CHR$(24)
370    KEY (7),CHR$(6)
380    KEY (8),CHR$(7)
390    KEY (9),CHR$(16)
400    KEY (10),CHR$(23)
410    CREV 1
420    FOR Z=0 TO 9
430        READ Z$:LOCATE Z*7,24:PRINT Z$;
440    NEXT Z
450    CREV 0:LOCATE 34,24:PRINT "|";
460    DATA 行挿入,行削除,上複写,下連結,改　行
470    DATA Cut&Pa,10行↓,10行↑,表示修,TxtBOX
480 REM ---------- 初期設定；
490    COLOR 5:CREV 1:LOCATE 0,0:PRINT CHR$(5);
500    PRINT TAB(10);"*  Ｓｕｐｅｒものかきくん：初期設定  *";
510    COLOR 7:CREV 0
520    LOCATE 39,3:PRINT "2"
530    LOCATE 0,3:INPUT "# 文書ＦＤのドライブ番号は（ 1 or 2 ）: ",FD$
540      IF FD$<>"2" THEN FD$="1"
550    FD$=FD$+":":CHDIR FD$
560    LOCATE 39,6:PRINT "80"
570    LOCATE 0,6:INPUT "# 桁数はいくらにしますか  （ 1 - 80 ）: ",XM
580    LOCATE 39,9:PRINT "300"
590    LOCATE 0,9:INPUT "# 行数はいくらにしますか  （ 1 -·300 ）: ",YM
600    COLOR 5:CREV 1:LOCATE 0,12:PRINT CHR$(5);
610    COLOR 6:CREV 0
620    LOCATE 10,15:INPUT "# いいですか？（Ｙ／Ｎ）: ",ANS$
630      IF ANS$="N" OR ANS$="n" OR ANS$="Ｎ" OR ANS$="ｎ" THEN 510
640    DIM W$(1,YM+1),V$(10,50),BAFA$(100)
650    CLS 2
660    FOR ZY=1 TO 20:FOR ZX=0 TO XM/2:PSET (ZX*8,ZY*8),9 :NEXT:NEXT
670 REM ---------- main();
680    CLS:COLOR 7:CREV 0
690    PRINT TAB(10);"＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊"
700    PRINT TAB(10);"＊ 新概念・日本語ワードプロセッサ    ＊"
710    PRINT TAB(10);"＊  ◆ Ｓｕｐｅｒものかきくん ◆    ＊"
720    PRINT TAB(10);"＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊＊"
730    COLOR 4:CREV 1
740    LOCATE 15,5:PRINT " ＜＜ ＭＥＮＵ ＞＞ "
750    LOCATE 15,21:PRINT " [CR]で決定／ANY=選択 "
760    PAUSE 5
770    CREV@ (15,6)-(36,20),1
780    COLOR@ (15,6)-(36,20),5
790    COLOR 7
800    FOR Z=1 TO 7:GOSUB 880:NEXT Z
810    Z=1
820    COLOR 6:GOSUB 880
830    Z$=INKEY$:IF Z$="" THEN 830
840      IF Z$=CHR$(13) THEN 970
850    COLOR 7:GOSUB 880
860    Z=Z+1:IF Z>7 THEN Z=1
870    GOTO 820
880    LOCATE 16,Z*2+5
890    ON Z GOSUB 900,910,920,930,940,950,960:RETURN
900    PRINT " Ｅ Ｄ Ｉ Ｔ     ":RETURN
910    PRINT " Ｌ Ｏ Ａ Ｄ     ":RETURN
920    PRINT " Ｓ Ａ Ｖ Ｅ     ":RETURN
930    PRINT " Ｐ Ｒ Ｉ Ｎ Ｔ  ":RETURN
940    PRINT " Ｄ Ｅ Ｌ Ｅ Ｔ Ｅ ":RETURN
950    PRINT " Ｍ Ｅ Ｒ Ｇ Ｅ   ":RETURN
960    PRINT " Ｔｈｅ ＥＮＤ    ":RETURN
970    COLOR 7:CREV 0
980    ON Z GOTO 1000,5090,5500,5690,5950,6070
990    CREV 0:COLOR 7:END
1000 REM ========== 編集を開始！
1010    CLOSE:OPEN "O",#3,"CRT1:CG"
1020    X=1:Y=1:YS=1:YR=1
1030    GOSUB 1240
1040    GOSUB 1100
1050      IF YE<YR THEN YE=YR
1060    FOR Z=1 TO 19
1070        LOCATE 0,Z:PRINT #3,W$(PG,YS+Z-1);
1080    NEXT
1090    GOTO 1290
1100  sub ---------- 画面の初期設定
1110      COLOR 7:CREV 0:CLS
1120      LOCATE 0,0:PRINT "--------+-<< Ｓｕｐｅｒものかきくん >>-+";
1130               PRINT "---------+---------+---------+---------";
1140      IF PG=1 THEN
```

編集した文章を本文に戻すときにはカット&ペーストの「MERGE」を選択して，TEXT-BOX MENUから合成する文章を選択します。

テキストボックスの文章は「長文の要素」であり，「メモカードに書き込んだ短い文章」です。好きなように位置づけて，活用してください。

また2つの文書ファイル間でも整理整頓ができるように，メインメニューのMERGE（文書合成）では合成する文書の内容を目で見て確認する機能を付けてみました。ちょっとしたことですが，便利です。

パソコン通信に最適

パソコン通信ネットワークをやっていますか？ そろそろ実用的なサービスが出てきているので，まだの人にはお勧めします。

さて，このSuperものかきくんは通常の文書作成に向くだけでなく，パソコン通信ネットワークでダウンロードした文書ファイルを整理するのに最適だという特徴があります。

ダウンロード文書ファイルとはパソコン通信における，あるポイントからあるポイントまでの画面に表示された文書や文字がすべて記録されたテキストファイルです。したがって，残しておいてあとで読みたいファイルもありますが，大部分は捨てるべき部分であったり，コマンドであったり，システムメニューであったりします。

こうしたファイルの整理は至難の技で，「あとでやろっと」なんて思ってとっておくと，ドンドンとたまる一方で，二度と見ることのない死蔵ファイルになることも決して珍しくありません。

ここでSuperものかきくんを使用すると，ダウンロードファイルから必要な部分だけを抜き取って，テキストボックスに収納して，それぞれを「裏の編集画面」で編集したうえで整理できます。そうやって整理していき，残りの「くず」はまとめて削除します。

こうして抽出した小さなファイルリストはTEXT-BOX MENUに蓄積されますが，そのときの表示画面はパソコン通信で実際に表示されるリストと同じフォーマットですので，思わずニヤリとすることうけあいです(図2)。

数少ないネットワーカーとして活躍(?)している私自身，テレホンソフトとのセットで毎日使っています。ネットワーカーの方にはぜひ利用をお勧めします。

```
1150         CREV 1:COLOR 6
1160         LOCATE 40,0:PRINT "！！ これは裏の編集画面です ！！";
1170         CREV 0:COLOR 7
1180      END IF
1190      LOCATE 0,20:PRINT " - - - - 1 - - - - 2 - - - - 3 - - - - 4";
1200                  PRINT " - - - - 5 - - - - 6 - - - - 7 - - - - ";
1210      COLOR 5
1220      C$=" ":C=0:LOCATE 15,21
1230   RETURN
1240   sub ---------- YE（最終行）を求める；
1250      FOR Z=YM TO 1 STEP -1
1260            IF W$(PG,Z)<>"" THEN YE=Z:GOTO 1280
1270      NEXT Z:YE=1
1280   RETURN
1290 REM ---------- ＩＮＫＥＹ＄ ；
1300    LOCATE 0,21:PRINT CHR$(26);"ｶｰｿﾙ:" X;YR
1310    C$=MID$(W$(PG,YR),X,2):IF C$="" THEN C$="  "
1320    CREV 1:COLOR 7:LOCATE X-1,Y:PRINT #3,C$;
1330    K$=INKEY$:IF K$="" THEN 1330
1340    CREV 0:COLOR 5:LOCATE X-1,Y:PRINT #3,C$;
1350    K=ASC(K$)
1360      IF K>31 THEN 1710
1370      IF K<28 THEN 1880
1380    IF K=LT THEN
1390       IF YR=1 AND X<2 THEN 1290
1400       X=X-2
1410       IF X<1 THEN X=XM-1:YR=YR-1:Y=Y-1
1420    END IF
1430    IF K=RT THEN
1440       IF YR=YM THEN 1290
1450       X=X+2
1460       IF X>XM THEN X=1:YR=YR+1:Y=Y+1
1470    END IF
1480    IF K=UP THEN
1490       IF YR=1 THEN 1290
1500       Y=Y-1:YR=YR-1
1510    END IF
1520    IF K=DN THEN
1530       IF YR=YM THEN 1290
1540       Y=Y+1:YR=YR+1
1550    END IF
1560 REM ---------- 境界条件；
1570    IF YR>YE THEN YE=YR
1580    IF Y>0 AND Y<20 THEN 1290
1590    CONSOLE 1,19
1600    IF Y<1 THEN
1610           Y=1:YS=YS-1:IF YS<1 THEN YS=1
1620           LOCATE 0,1:PRINT CHR$(15);:PRINT #3,W$(PG,YR);
1630    END IF
1640    IF Y>19 THEN
1650            IF YR>YM THEN 1050
1660            Y=19:YS=YS+1:IF YS>YM-19 THEN YS=YM-19
1670            LOCATE 0,19:PRINT:LOCATE 0,19:PRINT #3,W$(PG,YR);
1680    END IF
1690    CONSOLE 0,24
1700    GOTO 1290
1710 REM ---------- 文字入力です；
1720    Z$=INKEY$
1730      IF Z$<>"" THEN K$=K$+Z$:GOTO 1720
1740    W$=W$(PG,YR)
1750      IF LEN(W$)<X-1 THEN W$=LEFT$((W$+SPACE$(X)),X-1)
1760    W$=LEFT$(W$,X-1)+K$+MID$(W$,X,255)
1770    W$(PG,YR)=LEFT$(W$,XM):W$=MID$(W$,XM+1,255)
1780    X=X+LEN(K$)
1790      IF W$="" THEN 1850
1800    YR=YR+1
1810    GOSUB 2910
1820    YR=YR-1
1830      IF X>XM THEN X=X-XM:Y=Y+1:YR=YR+1
1840    GOTO 1040
1850    LOCATE 0,Y:PRINT SPACE$(XM);
1860    LOCATE 0,Y:PRINT #3,W$(PG,YR);
1870    GOTO 1560
1880 REM ---------- 編集キーが押されたとき・・・；
1890    IF K=1 THEN GOSUB 2360:GOTO 1040:REM ------ Lins
1900    IF K=22 THEN 2430:REM ------ 1行削除
1910    IF K=3 THEN 2570:REM ------ 1行上複写
1920    IF K=4 THEN 2240:REM ----- 1文字削除
1930    IF K=5 THEN 2510:REM ----- afterclear
1940    IF K=7 THEN 2990:REM ----- 行揃え
1950    IF K=6 THEN 2830:REM ----- 10行前進
1960    IF K=2 THEN 2870:REM ----- 10行後退
1970    IF K=8 THEN 2240:REM ----- 1文字削除
1980    IF K=9 THEN 2740:REM ----- word
1990    IF K=10 THEN 2630:REM ----- joint
2000    IF K=11 THEN 1000:REM ----- 文頭
2010    IF K=12 THEN FOR Z=YR TO YE:W$(PG,Z)="":NEXT:
          YE=YR:X=1:GOTO 1040
2020    IF K=13 THEN X=1:Y=Y+1:YR=YR+1
2030    IF K=16 THEN 1850:REM ----- 再表示
2040    IF K=18 THEN 2310:REM ----- ins
2050    IF K=17 THEN 2110:REM ----- 記号
2060    IF K=23 THEN 4720:REM ----- TEXT-BOX
2070    IF K=24 THEN 3070:REM ----- Unit of Sentence / INDEX
2080    IF K=19 THEN 2680:REM ----- Separate
2090    IF K=27 THEN 670:REM ----- escape
2100    GOTO 1560
2110 REM ---------- 記号選択；
2120    LOCATE 0,21:PRINT CHR$(26);
2130    FOR Z%=0 TO 15:GOSUB 2230:NEXT:Z%=0
2140    CREV 1:GOSUB 2230
2150    K$=INKEY$:IF K$="" THEN 2150
2160    CREV 0:GOSUB 2230
2170    IF K$=CHR$(29) THEN Z%=Z%-1:IF Z%<0 THEN MK%=MK%-16:Z%=15 ELSE 2140
2180    IF K$=CHR$(28) THEN Z%=Z%+1:IF Z%>15 THEN MK%=MK%+16:Z%=0 ELSE 2140
2190    IF K$=CHR$(30) THEN MK%=MK%-16
```

マニュアル編

動作環境

動作させるために最低限必要なものは次のとおりです。

▽ハード＝MZ-2500シリーズ(主記憶容量256Kバイト，グラフィックRAM1ページ，辞書ROM，ディスクドライブ1台以上)

▽基本ソフト＝BASIC-M25/V 2 (できればV 2がいい)

早い話がMZ-2531に相当する環境があればすべての機能を使うことができます。やはりMZ-2500は初めからV 2であるべきであった。

プログラムの構成は次のとおりです。

- 初期設定
  - ・EDIT（文書作成・編集）
  - ・LOAD（文書読み出し）
  - ・SAVE （文書書き込み）
  - ・PRINT（文書印刷）
  - ・DELETE（文書削除）
  - ・MERGE（文書合成）
  - ・The END

初期設定

起動後に初期設定用のテーブルが表示されます。用意した項目は

・文書FDドライブの指定

・1行の文字の桁数（最大漢字40字）

・行数（制限なし）

の3つです。行数ですが，あまり長くなると処理速度が遅くなってしまいますので，案内書や回覧文などは100行程度に，長い文書は300行から1000行にと，ケース・バイ・ケースで指定してください。

日本語処理機能

MZ-2500V 2用BASICの日本語処理機能をそのまま使用しています。かな漢字変換は，内蔵の辞書ROMを使った複合語単位の文節変換です。

半角/全角の切り換えは［SHIFT］＋［変換］。変換モードは［変換］キーを押すことで，文節変換から単漢字，人名・地名，コード入力，半角の切り換えです。

BASICがV 2バージョンですと，文節変換時に学習機能が使え，ユーザー辞書も使用できます。

漢字入力モードでの機能の切り換えは

```
2200    IF K$=CHR$(31) THEN MK%=MK%+16
2210    IF K$=CHR$(13) THEN MK%=MK%+Z%:K$=CHR$(MK%):GOTO 1740
2220    GOTO 2110
2230    LOCATE Z%*4,21:PRINT CHR$(MK%+Z%):RETURN
2240 REM ========== 1文字削除
2250    W$=W$(PG,YR)
2260    K$=MID$(W$,X,1):K=ASC(K$)
2270      IF (K>&H7F AND K<&HA0) OR K>&HDF THEN K$=MID$(W$,X,2)
2280    W$=LEFT$(W$,X-1)+MID$(W$,X+LEN(K$),XM)
2290    W$(PG,YR)=W$
2300    GOTO 1850
2310 REM ========== 空白文字挿入
2320    W$=W$(PG,YR)
2330    W$=LEFT$(W$,X-1)+" "+MID$(W$,X,XM)
2340    X=X-1
2350    GOTO 1770
2360   sub --------- 1行挿入
2370    FOR Z=YE TO YR STEP -1
2380        W$(PG,Z+1)=W$(PG,Z)
2390    NEXT Z
2400    YE=YE+1:IF YE>YM THEN YE=YM
2410    W$(PG,YR)="":X=1
2420   RETURN
2430 REM ========== 1行削除
2440    Y0=YR:Y1=YR
2450    FOR Z=YR TO YE-1
2460        W$(PG,Z)=W$(PG,Z+1)
2470    NEXT Z
2480    W$(PG,YE)=""
2490    YE=YE-1:IF YE<1 THEN YE=1
2500    X=1:GOTO 1040
2510 REM ========== アフタークリア；
2520    W$(PG,YR)=LEFT$(W$(PG,YR),X-1)
2530    LOCATE 0,Y
2540    PRINT #3,LEFT$((W$(PG,YR)+SPACE$(XM)),XM);
2550    LOCATE X,Y
2560    GOTO 1290
2570 REM ========== 1行上のコピー
2580    W$=W$(PG,YR)+SPACE$(80)
2590    W$(PG,YR)=LEFT$(W$,X-1)+MID$(W$(PG,YR-1),X,255)
2600    COLOR 6:LOCATE X-1,Y:PRINT #3,MID$(W$(PG,YR-1),X,255)
2610    COLOR 5:LOCATE X-1,Y:PRINT #3,MID$(W$(PG,YR-1),X,255)
2620    GOTO 1290
2630 REM ========== 次の行との連結
2640    W$=W$(PG,YR)+W$(PG,YR+1)
2650    W$(PG,YR)=LEFT$(W$,XM)
2660    W$(PG,YR+1)=MID$(W$,XM+1,255)
2670    GOTO 1040
2680 REM ========== カーソル後改行
2690    W$=W$(PG,YR)
2700    W$(PG,YR)=LEFT$(W$,X-1)
2710    W$=MID$(W$,X,255)
2720    YR=YR+1:GOSUB 2910:YR=YR-1
2730    GOTO 1040
2740 REM ========== タビュレーション
2750    W$=W$(PG,YR):W=LEN(W$)
2760    IF X>=W THEN 1560
2770    Z=X+1
2780    WHILE (MID$(W$,Z-1,1)<>" ") OR (MID$(W$,Z,1)=" ")
2790          Z=Z+1:IF Z>W THEN 2810
2800    WEND
2810    X=Z:IF X>XM THEN X=XM
2820    GOTO 1560
2830 REM ========== 10行前進
2840    YR=YR+10:YS=YS+10
2850    IF YS+18>YM THEN YS=YM-18:YR=YM:Y=19
2860    GOTO 1040
2870 REM ========== 10行後退
2880    YR=YR-10:YS=YS-10
2890      IF YR<1 THEN YR=1:Y=1:YS=1
2900    GOTO 1040
2910  sub ---------- 文字があふれたときのサブルーチン
2920    ZY=YR
2930    IF LEFT$(W$(PG,ZY),1)=" " THEN YY=YR:YR=ZY:GOSUB 2360:YR=YY
2940    ZW$=W$+W$(PG,ZY)
2950    W$(PG,ZY)=LEFT$(ZW$,XM)
2960    W$=MID$(ZW$,XM+1,255)
2970      IF W$="" THEN RETURN
2980    ZY=ZY+1:GOTO 2930
2990 REM ========== 行揃え
3000     FOR Z=1 TO YE-1
3010           IF LEFT$(W$(PG,Z+1),1)=" " THEN 3050
3020         W$=W$(PG,Z)+W$(PG,Z+1)
3030         W$(PG,Z)=LEFT$(W$,XM)
3040         W$(PG,Z+1)=MID$(W$,XM+1,255)
3050     NEXT Z
3060     GOTO 1040
3070 REM ========== Superカット&ペースト
3080    Y0=YR:Y1=YR:'    Y0:開始行   Y1:最終行
3090    YB=Y:YRB=YR:YSB=YS
3100    GOSUB 1100
3110    CREV 0:COLOR 5
3120    FOR Z=1 TO 19:ZY=YS+Z-1
3130        IF (ZY>=Y0)*(ZY<=Y1) THEN CREV 1:COLOR 6:LOCATE 0,Z:PRINT TAB(XM);
3140        LOCATE 0,Z:PRINT #3,W$(PG,ZY);
3150        CREV 0:COLOR 5
3160    NEXT
3170    LOCATE 0,21:PRINT CHR$(26);"ｶｰｿﾙ:" X;YR
3180    K$=INKEY$:IF K$="" THEN 3180
3190      IF YE<YR THEN YE=YR
3200    K=ASC(K$)
3210      IF K=13 THEN 3390
3220      IF (K<30) OR (K>31) THEN 3180
3230    IF K=UP THEN
3240          IF YR=Y0 THEN PRINT CHR$(7);:GOTO 3180
```

［F1］かな/ローマ字入力
［F2］ひらがな/カタカナ
［F5］入力したひらがな/カタカナ反転
［カナ］かな/英数字
［LOCK］大文字/小文字

となっています。さらにこのほかにも［F10］を押すと，直前に入力した文字列を再入力できるなどの機能を備えていますが，詳しくはMZ-2500のマニュアルを参照してください。

基本編集機能

フルスクリーンエディタです。カーソルは4方向の矢印キーで，上下左右に動き当然ながら画面は上下にスクロールします。

では今回のバージョンで用意した編集機能を機能別にざっと説明しましょう。

- 1文字削除：［CTRL］＋［D］
- 1文字挿入：［INST］
- 10行前進：［F7］または［CTRL］＋［F］
- 10行後退：［F8］または［CTRL］＋［B］
- 文頭移動：［HOME］
- 次の文節にカーソル移動：［TAB］
- カーソルのみの改行：［CR］
- 1行挿入：［F1］または［CTRL］＋［A］
- 1行削除：［F2］または［CTRL］＋［V］
- 1行複写：［F3］または［CTRL］＋［C］
- 下行連結：［F4］または［CTRL］＋［J］
- カーソル以降の改行：［F5］または［CTRL］＋［S］
- アフタークリア：［CTRL］＋［E］（カーソル右側の文字列削除）
- アンダークリア：［CLR］

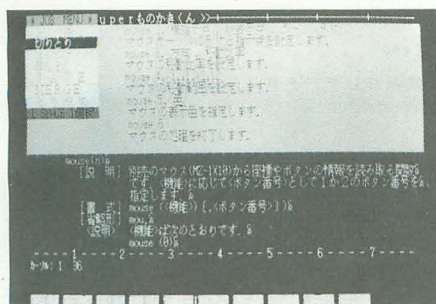
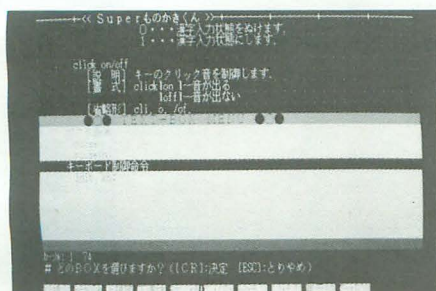
カット＆ペースト

まずSuperものかきくんの特徴のひとつ目がこれです。複数行をまとめて一気に効率よく編集ができるのです。操作手順は次のとおりです。

1) ［CTRL］＋［X］または［F6］を押す。
2) 対象範囲が黄色でリバース表示されます。上下矢印キーで範囲を広げてからリターンキーで決定します。
3) すると美しいプルダウンメニューが表示されます。ここから機能を選択してリターンキーで決定します。

プルダウンメニューに装備した機能は以下の7つです。

- 空白行挿入：対象範囲と同じ長さの行を

```
3250       LOCATE 0,Y:PRINT TAB(XM);
3260       LOCATE 0,Y:PRINT #3,W$(PG,YR)
3270       Y=Y-1:YR=YR-1:Y1=Y1-1
3280         IF Y<1 THEN YS=YS-1:Y=1:GOTO 3100
3290    END IF
3300    IF K<>DN THEN 3170
3310    Y=Y+1:YR=YR+1:Y1=YR
3320      IF YR>YM THEN Y=19:YS=YM-18:YR=YM
3330      IF Y>19 THEN Y=19:YS=YS+10:YR=YR+9:Y1=YR:GOTO 3100
3340    CREV 1:COLOR 6
3350    LOCATE 0,Y:PRINT TAB(XM);
3360    LOCATE 0,Y:PRINT #3,W$(PG,YR);
3370    CREV 0:COLOR 5
3380    GOTO 3170
3390 REM ---------- プルダウンメニュー；
3400    CONSOLE 0,9,0,15
3410    Y1=YR
3420    Y=YB:YS=YSB:YR=YRB
3430    COLOR 2:CREV 1
3440    LOCATE 0,0:PRINT " * JOB  MENU * ";CHR$(26)
3450    COLOR 7
3460    FOR Z=1 TO 7
3470        LOCATE 0,Z:GOSUB 3580
3480    NEXT Z
3490    COLOR 2
3500    LOCATE 0,8:PRINT " [ SPACE ]選択";
3510    COLOR 7:Z=1
3520    CREV 0:LOCATE 0,Z:GOSUB 3580
3530    K$=INKEY$:IF K$="" THEN 3530
3540       IF K$=CHR$(13) THEN 3660
3550    CREV 1:LOCATE 0,Z:GOSUB 3580
3560    Z=Z+1:IF Z>7 THEN Z=1
3570    GOTO 3520
3580    ON Z GOSUB 3590,3600,3610,3620,3630,3640,3650:RETURN
3590       PRINT "  空白行挿入   ";:RETURN
3600       PRINT "  切りとり     ";:RETURN
3610       PRINT "  はりつけ     ";:RETURN
3620       PRINT "  写しとり     ";:RETURN
3630       PRINT "  S A V E    ";:RETURN
3640       PRINT "  M E R G E   ";:RETURN
3650       PRINT "  と り や め   ";:RETURN
3660    CONSOLE 0,24,0,80
3670    ON Z GOTO 3760,3890,3990,4100,4120,4610,1040
3680    GOTO 1000
3690 REM
3700 REM ---------- BAFA'S COPY:Y 0行からY 1行までをとりこむ
3710    FOR Z=Y0 TO Y1
3720        BAFA$(Z-Y0+1)=W$(PG,Z)
3730    NEXT Z
3740    LBAFA=Y1-Y0+1
3750    RETURN
3760 REM ---------- 挿  入：
3770     GOSUB 3780:GOTO 1040
3780     LINST=Y1-Y0+1
3790     FOR Z=YE TO Y0 STEP -1
3800         IF Z+LINST>YM THEN 3820
3810         W$(PG,Z+LINST)=W$(PG,Z)
3820     NEXT Z
3830     FOR Z=Y0 TO Y1
3840         IF Z>YM THEN 3870
3850         W$(PG,Z)=""
3860     NEXT
3870     YE=YE+LINST:IF YE>YM THEN YE=YM
3880     RETURN
3890 REM ---------- C U T ;
3900    GOSUB 3700
3910    FOR Z=Y0 TO YE-LBAFA
3920        W$(PG,Z)=W$(PG,Z+LBAFA)
3930    NEXT Z
3940    FOR Z=YE-LBAFA+1 TO YE
3950        W$(PG,Z)=""
3960    NEXT
3970    YE=YE-LBAFA
3980 GOTO 1040
3990 REM ---------- ペースト；
4000        IF Z=0 THEN 1040
4010     LINST=LBAFA:IF LINST=0 THEN 1040
4020     Y0=YR:Y1=Y0+LBAFA-1
4030     GOSUB 3790
4040     FOR Z=Y0 TO Y1
4050         IF Z>YM THEN 4080
4060         W$(PG,Z)=BAFA$(Z-Y0+1)
4070     NEXT Z
4080       IF YE>YM THEN YE=YM
4090     GOTO 1040
4100 REM ---------- C O P Y ;
4110     GOSUB 3700:GOTO 1040
4120 REM ---------- L O C A L  S A V E ;
4130     COLOR 4:CREV 1:LOCATE 18,5:PRINT "* ローカルSAVE *"
4140     COLOR 7:CREV 1
4150     FOR Z=1 TO 3
4160         ON Z GOSUB 4270,4280,4290
4170     NEXT:Z=3
4180     COLOR 4:LOCATE 18,9:PRINT " ANY-SELECT/[CR]:決定 "
4190     COLOR 7
4200     CREV 0:ON Z GOSUB 4270,4280,4290
4210     Z$=INKEY$:IF Z$="" THEN 4210
4220     IF Z$=CHR$(13) THEN 4260
4230     CREV 1:ON Z GOSUB 4270,4280,4290
4240     Z=Z+1:IF Z>3 THEN Z=1
4250     GOTO 4190
4260     ON Z GOTO 4470,4300,1040
4270     LOCATE 18,6:PRINT " D I S K - F I L Eに  ";:RETURN
4280     LOCATE 18,7:PRINT " T E X T - B O X  に  ";:RETURN
4290     LOCATE 18,8:PRINT "   と  り  や  め     ";:RETURN
```

プルダウンメニューによるカット&ペースト

便利なテキストボックス

　　　　挿入します。

・切りとり：対象範囲を削除します。

・はりつけ：カットバッファの内容を挿入します。

・写しとり：対象範囲を削除せず，カットバッファにコピーします。

・SAVE　：対象範囲だけを切り出して，テキストボックスかフロッピーにセーブします。

・MERGE：テキストボックスに登録してある文章を挿入します。

・とりやめ：作業中止

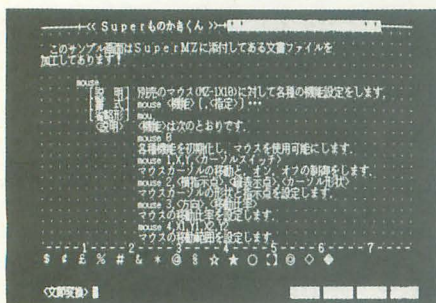
「移動」は用意しませんでしたが，「切りとり」と「はりつけ」を組み合わせることで可能です。

テキストボックスの操作方法

登録　［CTRL］＋［X］または［F6］でプルダウンメニューを呼び出して，［SAVE］を選択したのち，［TEXT-BOX MENU］を選んで10個の引き出しのうち，好きなところでリターンキーを押します。

ボックス内編集　［CTRL］+［W］または［F10］を選び，［TEXT-BOX MENU］を選んで10個の引き出しのうち，好きなところでリターンキーを押します。これで「裏の編集画面」での編集に入ります。終了したら再び［CTRL］＋［W］または［F10］です。

ボックス内文書のマージ　［F6］または［CTRL］＋［X］で挿入位置を指定したのち，プルダウンメニューで［MERGE］を選択します。

```
4300 REM ---------- TEXT-BOXに転送；
4310   IF Y1-Y0+1<=VM THEN 4350
4320  COLOR 6:LOCATE 0,22
4330  PRINT "！！ カットした行数が規定値の";VM;"行を超えています ！！"
4340  FOR Z=1 TO 999:NEXT:CREV 0:GOTO 1040
4350  COLOR 6:LOCATE 0,22
4360  PRINT "# どのウインドーに保管しますか？ （[CR]:決定）";
4370  GOSUB 4890
4380  FOR Z=Y0 TO Y1
4390    IF Y1-Y0>50 THEN 3810
4400    V$(WN,Z-Y0+1)=W$(PG,Z)
4410  NEXT Z
4420  COLOR 6:CREV 0
4430  LOCATE 0,22:PRINT CHR$(26);
4440  INPUT "# この部分を削除しますか？(Y/N):",ANS$
4450   IF ANS$="Y" OR ANS$="y" OR ANS$="Ｙ" OR ANS$="ｙ" THEN 3890
4460  GOTO 1040
4470 REM ---------- フィールドファイルの生成：
4480  COLOR 6:LOCATE 0,21
4490  PRINT CHR$(5);"# この部分だけのファイルをつくります．"
4500  PRINT CHR$(5);:INPUT "  ファイル名は（［.］でとりやめ）:";ANS$
4510   IF ANS$="" OR ANS$="." OR ANS$="．" THEN 1040
4520  ON ERROR GOTO 4590
4530  OPEN "O",#1,ANS$
4540   FOR Z=Y0 TO Y1
4550     PRINT #1,W$(PG,Z)
4560   NEXT Z
4570  CLOSE #1
4580  GOTO 4420
4590   IF ERR=42 AND ERL=5610 THEN KILL F$:GOTO 4520
4600 STOP
4610 REM ---------- ＬＯＣＡＬ ＭＥＲＧＥ；
4620   IF PG<>0 THEN 4810
4630  COLOR 6:LOCATE 0,22
4640  PRINT "# どのＢＯＸを選びますか？（［ＣＲ］：決定）"
4650  GOSUB 4890
4660  FOR Z=50 TO 1 STEP -1
4670   IF V$(WN,Z)<>"" THEN 4690
4680  NEXT
4690  Y1=Y0+Z:GOSUB 3780
4700  FOR ZZ=1 TO Y1-Y0:W$(PG,Y0+ZZ-1)=V$(WN,ZZ):NEXT
4710  GOTO 1030
4720 REM ---------- ＴＥＸＴ－ＢＯＸ；
4730   IF PG<>0 THEN 4810
4740  COLOR 6:LOCATE 0,22
4750  PRINT "# どのＢＯＸを選びますか？（[ＣＲ]:決定  [ESC]:とりやめ）"
4760  GOSUB 4890
4770  PG=1
4780  FOR Z=1 TO 50:W$(PG,Z)=V$(WN,Z):NEXT
4790  FOR Z=1 TO 50:V$(WN,Z)="":NEXT
4800  GOTO 1000
4810 REM ========== 本体への復帰
4820 COLOR 6:LOCATE 0,22
4830 PRINT "# どのＢＯＸに保管しますか？（［ＣＲ］：決定）"
4840 GOSUB 4890
4850 FOR Z=1 TO 50:V$(WN,Z)=W$(PG,Z):NEXT
4860 FOR Z=1 TO YM:W$(PG,Z)="":NEXT
4870 PG=0
4880 GOTO 1000
4890 ' sub ---------- ＢＯＸ－ＬＩＳＴ
4900  CREV 1:COLOR 4
4910  LOCATE 0,9:PRINT TAB(10);
4920  PRINT "●○●  ＴＥＸＴ－ＢＯＸ  ＭＥＮＵ  ●○●";SPC(32);
4930  CREV 0:COLOR 7
4940  FOR Z=1 TO 10
4950      LOCATE 0,9+Z:PRINT CHR$(5);V$(Z,1);
4960  NEXT Z
4970  COLOR 4:CREV 1
4980  LOCATE 0,20:PRINT SPACE$(80);
4990  WN=1:COLOR 7
5000  CREV 1
5010  LOCATE 0,9+WN:PRINT CHR$(5);V$(WN,1)
5020  WN$=INKEY$:IF WN$="" THEN 5020
5030    IF WN$=CHR$(13) THEN RETURN
5040    IF WN$=CHR$(27) THEN RETURN
5050  CREV 0
5060  LOCATE 0,9+WN:PRINT CHR$(5);V$(WN,1);
5070  WN=WN+1:IF WN>10 THEN WN=1
5080  GOTO 5000
5090 REM ---------- ＬＯＡＤＩＮＧ：
5100 DF$="":F$="":ANS$=""
5110 CHDIR:CLS
5120 CREV 1:COLOR 5
5130 PRINT CHR$(5);" ＜＜＜ ＬＯＡＤＩＮＧ ＞＞＞"
5140 CONSOLE 1,23
5150 CREV 0:COLOR 7
5160 FILES
5170 CREV 1:COLOR 5
5180 PRINT CHR$(5)
5190 CREV 0:COLOR 7
5200 PRINT:PRINT:INPUT "# input FILE NAME ([B]ack/[C]ancel):";F$
5210   IF (F$="B") OR (F$="b") THEN 5150
5220   IF (F$="C") OR (F$="c") THEN 5490
5230 PRINT:INPUT " 階層型FILEですか (Y/N):";ANS$
5240 IF ANS$="Y" OR ANS$="Ｙ" THEN
5250    CHDIR F$
5260    CLS:PRINT "----- DIR:";F$;" -----"
5270    PRINT:FILES
5280    DF$=DF$+"/"+F$+"/"
5290    GOTO 5200
5300 END IF
5310 F$=DF$+F$
5320 PRINT     :INPUT " いくつか DATAを 読みとばしますか (Y/N):";ANS$
5330 OPEN "I",#1,F$
5340   IF (ANS$<>"y") AND (ANS$<>"Y") THEN 5400
```

これが噂の裏画面編集だ

TEXT-BOX MENUが表示されますので，マージする文章を選択してください。

ファイル管理

MZ-2500用標準シーケンシャルファイルをそのままテキストファイルとして使用しています。ですからアルゴエディタやテレホンソフトで作成したりダウンロードした文書はすべて使用できます。もちろんSuperものかきくんで作成した文書ファイルは他のソフトででも使えます。

ロード/セーブ/マージ/デリートとも対話式の操作になっていますので，プログラムのリクエストどおりに文書名を記入したり，YやNを入力していってください。

以下，手短にいくつか注意点を列記してみます。

・階層型ディレクトリに対応しています。内容ごととか目的ごとにファイルをまとめて管理してください。

・ドライブ番号は初期設定に依存します。別のドライブを設定するときには1：〜のように指定してください。

エラーが発生したら……

頑張って作ったのですが，まだ一部にバグが残っているかもしれません。もしも，不具合が生じましたときは，遠慮なく，[SHIFT] + [BREAK] を押して中断して

GOTO 1000 [CR]

と入力してください。いかなる状況でも回復させることができます。文書ファイルやテキストボックスの内容は完全に保護されていますのでご安心を。

＊　＊　＊

このプログラムは基本的にX1 turboにも移植不可能ではありません。自信のある方は挑戦してみてはいかがでしょうか。
まだ印刷機能など弱い部分がたくさんあります。私自身，今後も機能強化を続けていく予定ですが，プログラム自体はシンプルであり，BASICで書いてありますので，みなさんもお気づきの点がありましたら，どしどし機能強化してください。では，ごきげんよう。

```
5350       INPUT "  何個読みとばしますか:";DD
5360       FOR Y-1 TO DD
5370            IF EOF(1) THEN PRINT "* 読みだしデータ終了":GOTO 5480
5380          LINE INPUT #1,W$
5390       NEXT Y
5400       FOR Y-1 TO YM
5410           IF EOF(1) THEN 5480
5420         LINE INPUT #1,W$:W-LEN(W$)
5430                IF W<-XM THEN W$(PG,Y)-W$:GOTO 5470
5440              W$(PG,Y)-LEFT$(W$,XM)
5450              W$-MID$(W$,XM+1,255):W-LEN(W$)
5460              Y-Y+1:GOTO 5430
5470       NEXT Y
5480    CLOSE #1:YZ-Y
5490    CONSOLE 0,24:GOTO 670
5500 REM ---------- SAVING;
5510    ON ERROR GOTO 5670
5520    CLS:CREV 1:COLOR 5
5530    PRINT CHR$(5);"      <<<  SAVE  >>>"
5540    CREV 0:COLOR 7
5550    LOCATE 0,3:PRINT "% input FILE NAME (/DIR NAME/+)([.]-csl):"
5560    LOCATE 2,4:PRINT F$
5570    LOCATE 0,4:INPUT ANS$
5580     IF ANS$-"." THEN 670
5590    F$-FE$+ANS$
5600    GOSUB 1240
5610    OPEN "O",#1,F$
5620       FOR Y-1 TO YE
5630          PRINT #1,W$(PG,Y)
5640       NEXT Y
5650    CLOSE #1
5660    GOTO 670
5670    IF ERR-42 AND ERL-5610 THEN KILL F$:GOTO 5610
5680 STOP
5690 REM ---------- Print;
5700    GOSUB 1240
5710    CLS
5720    CREV 1:COLOR 5
5730    PRINT CHR$(5);"       <<< 文書印刷 >>>"
5740    CREV 0:COLOR 7
5750    PRINT:PRINT "* この文書の長さは";YE;"行あります。"
5760    LOCATE 21,5:PRINT "1"
5770    LOCATE 0,5:INPUT "* Print from Line No.",YS
5780    LOCATE 20,7:PRINT YE
5790    LOCATE 0,7:INPUT "            to   Line No.",YE
5800    LOCATE 0,9:INPUT "  TAB設定        (0-80):",PTAB
5810    LOCATE 0,11:INPUT "* 改行幅はいくらですか? (0/1):",HABA$
5820      IF HABA$-"" THEN HABA$-"0"
5830      IF HABA$<>"1" AND HABA$<>"0" THEN 5810
5840    CREV 1:COLOR 5
5850    LOCATE 0,13:PRINT CHR$(5)
5860    CREV 0:COLOR 7
5870    LOCATE 0,15:INPUT "# I am printing, OK (Y/N/E):";ANS$
5880                IF ANS$-"n" OR ANS$-"N" THEN 5770
5890                IF ANS$-"e" OR ANS$-"E" THEN 670
5900    FOR Y-YS TO YE
5910       LPRINT SPACE$(PTAB);W$(PG,Y)
5920         IF HABA$-"1" THEN LPRINT
5930    NEXT Y
5940    GOTO 670
5950 REM ---------- DELETING;
5960    CLS
5970    LOCATE 5,0:PRINT "--- D E L E T I N G ---"
5980    PRINT:FILES
5990    PRINT:PRINT:INPUT "# 削除するファイル名は (Quit-[.])";F$
6000      IF F$-"." THEN 670
6010    PRINT
6020    PRINT "% Now I'm deleting File '";F$;"'"
6030    INPUT "  いいですか (Y/N):";ANS$
6040      IF (ANS$<>"y") AND (ANS$<>"Y") THEN 670
6050    KILL F$
6060    GOTO 670
6070 REM ---------- MERGING;
6080    CLS:COLOR 5:CREV 1
6090    PRINT CHR$(5);TAB(10);"<<  MERGE  >>"
6100    COLOR 7:CREV 0
6110    ZL-1
6120    FOR Z-YE-4 TO YE:IF Z<1 THEN 6150
6130        LOCATE 0,ZL:PRINT W$(PG,Z)
6140        ZL-ZL+1
6150    NEXT Z
6160    LOCATE 0,6:PRINT STRING$(80,"-")
6170    LOCATE 9,9:PRINT "<<< merging file >>>"
6180    CONSOLE 10,14
6190    FILES:PRINT:CONSOLE 0,24
6200    COLOR 6:LOCATE 0,1:PRINT CHR$(5);
6210    INPUT "# ファイル名は (esc-[.]):",FZ$
6220    COLOR 7
6230      IF FZ$-"." THEN 6400
6240    LOCATE 0,7:PRINT CHR$(26)
6250    OPEN "I",#1,FZ$
6260        FOR Z-1 TO 10
6270              IF EOF(1) THEN 6300
6280            LINE INPUT #1,W$
6290            LOCATE 0,7+Z:PRINT W$
6300        NEXT Z:CLOSE #1
6310    COLOR 6
6320    LOCATE 0,20:INPUT "# OKですか? (again-[.]):",ANS$
6330      IF ANS$-"." THEN 6180
6340    OPEN "I",#1,FZ$
6350        FOR Z-ZZ+1 TO YM
6360              IF EOF(1) THEN 6390
6370            LINE INPUT #1,W$(PG,Z)
6380        NEXT Z
6390    CLOSE #1:YE-Z
6400    CONSOLE 0,24:GOTO 670
```

最終回
通信プログラムである

Iwai Ippei
祝　一平

私が講師の祝一平である。

まずは先にいっておくが，6，7月号でやったMMLで説明の足りない部分があったので補足しておく。

あのMMLは，G-RAMを演奏データ用のバッファとして使っているので，ctrl-D/INITなどによってグラフィックをONにすると，画面にバッファの内容が表れるのである。またそれに伴い，ループを組むなどして長い曲を演奏しようとすると，バッファが足りなくなってエラーが発生するようになっている。

次に打ち込み方であるが，うっかりして7月号ではノン・turboの場合の変更点を指摘するのを忘れてしまっていたのである。これは6月号の53ページにもあるように，turboでない機種では，

A965：A3　1F→07　07
A9D6：A0　1F→04　07
A9E5：A3　1F→07　07

というように変更しておいていただきたい。そしてじつにタコスケなことに，たっぷりとバグが出ている。ごめんなさいのページに従って直していただきたい。まことに申しわけない。

本題である

今月は何をするのかというと，これがなんと通信なのである。具体的にはCZ-8BM2(RS-232C・マウスボード）用のコントロールプログラムなのである。プログラムはturbo BASICのRS-232C関係の命令と同等のことができるようにすることを目的として作った。よってturboユーザーにとっては基本的におよびでない。しかしプログラム中のSIO，CTCのアドレスをずらし，さらにCZ-8FB01を使うか（もしくはCZ-8FB02であっても割り込みベクトルのつじつまを合わせる）すればturboでも使えることは使えるから，「BASICを使わない通信プログラム」を作る際の参考ぐらいにはなるであろう。

で，じつはCZ-8BM2にはカセットに入ったアプリケーションプログラム(機械語)が付いてくるのであるが，どうやら今月のプログラムはそれよりはましなものになったようである。ただし，ON COM GOSUBの受信割り込みはないので，いまいち使いづらいのは覚悟していただきたい。

ところで通信であるが，私の持論では「通信＝不毛」ということになっている。誰がなんといってもNTTが儲かるだけなのだ。第一，単なる民間企業のくせにテレホンカードで大儲けしているというのが気にくわん。また，日本ではぼちぼち通信速度が1200bps以上になろうとしているらしいが，はっきりいってそれでもタコなのである。もう少し“待ち”であろう。

繰り返すが，現状ではパソコン通信などをやってもNTTと一部業者を儲けさせるだけである。よって今月のプログラムは，できればパソコン通信などには使わないよーに。

それではいきなり始める。

X1のユーザーはturbo BASICのマニュアルを持っていないわけだから，表1（turboZのマニュアルから引用）を参考にしていただきたい。ただしプログラムはBASICのコマンドを拡張するわけではないので，USR0～USR9を使うことになっている。その際，文字列変数を渡すときには“その長さ”にちょいと気をつけなければいけないので，サンプルプログラムをよく見て使うように。特にUSR2，USR3が返す“有効文字数”は先頭のアスキーコードが示しているので，

A$=USR2(～)
A$=MID$(A$,2,ASC(LEFT$(A$,1)))

とすること。なお，SAVE，SAVEM，LOAD，LOADMはないので，自分で（BASICで）プログラムを組むなどして解決するよーに。

そしてリスト1＝ソースリスト，リスト2＝ダンプリスト，リスト3＝サンプルプログラム，リスト4＝簡単なターミナルプログラムである。リスト1の最初のほうを見るとわかるだろうが，Z80 SIO，CTCのアドレスがそれぞれ1F98Hと1FA8Hとなっている。これはCZ-8BM2のアドレス設定スイッチが，工場出荷時の2-3側になっている場合のものである。スイッチが1-2側もしくはturboの場合には，それぞれ1F90H，1FA0Hとすればよい。あとは説明してもむなしいのでやめておく。

よくわからないのである

ここで今月のプログラムの手抜きなどについて書いておくことにする。

まずはXON，XOFF制御についてである。やっていてだんだん面倒になってきたので，「送られてきたXON/XOFFはバッファにたまる」ようになっている。ただしバッファのトップがXONもしくはXOFFであったならば，「こちらからデータを送るときにチェックして捨てる」ようにしている。もしも向こうから「XON/XOFFとほかのコードも混ぜて送ってきたら」受信に混ざり込むことになる。必要ならば表示ルーチンなどで取り除いていただきたい。これは，バイナリデータとしてたまたまXON/XOFF（11H/13H）が送られてくる場合もあるのでこうしたのだが，本当はどうすべきだったのであろうか。またCLOSE時の動作もよくわかんないのである。turbo BASICでは“O”でオープンしたときだけエンドコードを送るようになっているようであるが，“C”では送らないようである。またその逆のエンドコードを受け取ったときの動作も不明瞭である。turbo BASICでは，LINPUTがエンドコードを受け取っ

表1　通信パラメータの説明

通信パラメータには次のものがある。
1) ボーレート
2) パリティ
3) データビット長
4) ストップビット長
5) 通信制御指定
6) カナの表現方法指定
7) CR，LFコードの送信処理
8) CR，LFコードの受信処理
9) 日本語文字列の表現方法指定（今月のプログラムでは“N”しか使えない）
10) エンドコード指定

1) ボーレート
0～6の数値によりボーレートを指定する。

数　値	0	1	2	3	4	5	6
ボーレート(ボー)	150	300	600	1200	2400	4800	9600

2) パリティ
E：偶数パリティチェックを使用する。
O：奇数パリティチェックを使用する。
N：パリティチェックを使用しない。

3) データビット長
5：データビット長を5ビットとする。
6：データビット長を6ビットとする。
7：データビット長を7ビットとする。
8：データビット長を8ビットとする。

4) ストップビット長
1：ストップビット長を1ビットとする。
2：ストップビット長を1.5ビットとする。
3：ストップビット長を2ビットとする。

5) 通信制御指定
X：XON/XOFFコードによる制御を行う。
R：RTS制御信号のON/OFFによる制御をする。
Nまたは省略：通信制御を行わない。
この通信制御とは，データ送受信時に受信側においてデータの受信処理が間に合わないとき，送信側に対して送信の一時停止を要求し受信できる状態になると送信再開を要求する方法のことをいう。
“X”を指定すると，データとしてXOFFコード（&H13）を送信側へ送ることによって送信の一時停止を要求し，XONコード（&H11）により送信再開を要求する。
“R”を指定すると，RTS信号線をOFFにすることによって送信の一時停止を要求し，ONにすることにより送信再開を要求する。

6) カナの表現方法指定
S：データビット長7ビットでカナの送受信ができる。
Nまたは省略：データビット長7ビットでカナの送受信ができない。
データビット長7ビットでのカナの送受信はシフトインコードSI（&H0F）があると，それ以降のデータをカナとして処理し，シフトアウトコードSO(&H0E)があればそれ以降のデータを英数字として処理する。

7) CR，LFコードの送信処理
C：復帰改行コードとしてCRコード（&H0D）を送信する。
Lまたは省略：復帰改行コードとしてCRコード(&H0D)＋LFコード（&H0A）を送信する。

8) CR，LFコードの受信処理
C：CRコード（&H0D）の受信によって復帰＋改行処理する。
Lまたは省略：CRコード（&H0D）のみを受信するとそれはデータとして処理される。CRコード(&H0D)＋LFコード（&H0A）を連続して受信すると，復帰＋改行処理をする。

9) 日本語文字列の表現方法指定
J：JIS漢字コードを使用する。漢字インコードKI（&H1B4B）で日本語文字列の始まりを示し，漢字アウトコードKO（&H1B48）で日本語文字列の終わりを示す。
Nまたは省略：シフトJIS漢字コードを使用する。
（今月のプログラムでは“N”しか使えない）

10) エンドコード指定
データ転送の終了を判断するためのエンドコードを指定する。エンドコードとしてコントロールコード（&H00～&H1F）の中から任意のひとつを選択できる。実際に指定するパラメータは，キャラクタコード&H40～&H5Fに対応する文字を使用する（&H60～&H7Fも可）。たとえば，D（またはd）と指定するとエンドコードとしてコントロールD（&H04）が使用される。またこのパラメータを省略すると，エンドコードの送受信処理は行われない。

〔注〕
・通信パラメータのうちボーレート，パリティ，データ長，ストップビット長は省略できないが，それ以降のパラメータは省略できる。ただし，途中のパラメータを省略して次のパラメータを指定することはできない。
・漢字の送受信はデータ長8ビットのときに可能(シフトJISコードを使うことになる)。また，カタカナの送受信はSI/SOコードを使うことによってデータ長が7ビットのときでも可能となる（カナの表現方法指定がSであれば自動的にカナに変換する)。

各USR関数について

USR0：OPEN	オープンする （パラメータ先頭に0，1，Cのいずれかを付加すること）
USR1：CLOSE	クローズする
USR2：LINPUT	1行入力する（CRもしくはCR＋LFまで）
USR3：INPUTN	N個の文字を入力する （個数は引数の文字列の先頭のアスキーコード）
USR4：PRINTS	文字列を出力する
USR5：PRINT	文字列＋CR(＋LF)を出力する
USR6：LOC	バッファ内のデータ数を返す
USR7：EOF	エンドコードを入力したかどうかを返す
USR8：PGETC1	生データを1文字入力する （カナの処理などはしない）
USR9：PPUTC1	生データを1文字出力する （カナの処理などはしない）

注）USR2，USR3が返す文字列の先頭のアスキーコードで入力されたデータ数を示す。

たときは1行が終了したとみなしてリターン，INPUT$で入力文字数を指定した場合はエンドコードがきたとしても，指定文字数までリターンしない，などとなっている。ここら辺の細かな動作は，BASICのマニュアルにきっちり書いてあるわけではないので，なんだかよくわかんないのである。

おっとここで思い出した。今月のプログラムでは，「EOFはこれから読み出そうとしているデータがエンドコードだったら－1を返す」ようになっている。よって不幸にしてINPUTNなどでその「エンドコードを読み出してしまったりすると」EOFでなくなってしまうのである。注意していただきたい。しかしLINPUTはエンドコードは読み出さないようになっているから，エンドコードが一度きたら，それ以降はずっとEOFは－1である。

なんだかんだ書いたが，はっきりいって結局はUSR8，USR9の「生データ読み出し，書き込み」が最終兵器となるような気がする。つらつら考えるに，この2つの機能＋LOC関数＋OPENコマンド＋あぶったイカさえあれば，ほかには何もいらないよーな気がしてしまうのである。ということは今月のプログラムは半分くらいが無駄だったのかもしれない。困ったことだ。

というわけで今月はこれで終わりである。

来月はないのである

来月は何をやるのかというと，おおっ！何もやることがないっ！　そう，目次およびタイトルからも明らかなように，この「試験に出るX1」は，今月をもってめでたく最終回なのである。このあと何かほかの連載でも始まるのかとか，立体視ボードはどーすんだとか，VHDコントロールボードの立場はどーなるんだとか，なんだなんだ試験はしねーのかだとか，いつまで待てば愛が地球を救うのだとか，24時間テレビよりも18時間×2日＝36時間テレビのほうが効果があるんじゃないかだとか，やっぱり中国は水爆ミサイルを持っているんだからあまり怒らせないほうがいいだとか，90年代のコンピュータなら90年代に入ってから作れだとか，最近の年寄りはなっとらん，わしらの若いころの年寄りはもっと控え目だったものぢゃだとか，どさくさにまぎれて関係ないことまで書くなだとか，いわれても困るのである。そこら辺のことは置いといて，とりあえずやることがなくなったので最終回なのである。というわけでまた会う日までである。

〈参考文献〉
額田忠之：Z80ファミリ・ハンドブック，CQ出版

リスト1　ソースリスト

```
                        .Z80
                        .PHASE  0EA00H
                ;
0330            QBREAK  EQU     0330H
1FCA            BBREAK  EQU     1FCAH
                ;
1F98            ZSIO    EQU     1F98H   ;IF NOT,1F90H
1FA8            ZCTC    EQU     1FA8H   ;IF NOT,1FA0H
                ;
005C            INTRSV  EQU     5CH     ;INT VECTOR
0040            QMAX    EQU     64      ;QUE SIZE
0080            QMAXX   EQU     QMAX+64 ;Q SIZE+ALPHA
                ;
000D            CR      EQU     0DH
000A            LF      EQU     0AH
000E            SO      EQU     0EH
000F            SI      EQU     0FH
0011            XON     EQU     11H
0013            XOFF    EQU     13H     ;CTRL CODES
                ;
                ;BAU RATE       0-6
                ;PARITY         E,O,N
                ;DATA BIT LEN   5,6,7,8
                ;STOP BIT LEN   1,2,3
                ;CONTROLE       X,R,N,""
                ;KANA MODE      S,N,""
                ;CR,LF  SEND    C,L,""
                ;CR,LF  RECEIVE C,L,""
                ;KANJI          J,N,""  ;DUMMY
                ;END CODE       '@'-'_' ;PARAMS
                ;
                ;JUMP TABLE
EA00    C3 EA1E         JP      OPEN    ;+0
                ;I,O or C + PARAMETER
EA03    C3 EB84         JP      CLOSE   ;+3
                ;TX ENDCODE
                ;
EA06    C3 ECDD         JP      LINPUT  ;+6
                ;UNTIL CR (+LF)
EA09    C3 ED35         JP      INPUTN  ;+9
                ;INPUT n CHAR
                ;
EA0C    C3 ED68         JP      PRINTS  ;+12
                ;TX STRING
EA0F    C3 ED56         JP      PRINT   ;+15
                ;TX STRING + CR (+LF)
                ;
EA12    C3 EB98         JP      LOC     ;+18
                ;RETURN BUFF SIZE
EA15    C3 EBB9         JP      EOF     ;+21
                ;EOF OR NOT
                ;
EA18    C3 EC9A         JP      PGETC1  ;+24
                ;RIMITIVE GET CHAR
EA1B    C3 ED90         JP      PPUTC1  ;+27
                ;PRIMITIVE PUT CHAR
                ;
EA1E    3A EE60 OPEN:   LD      A,(OPENF)
EA21    B7              OR      A
EA22    28 07           JR      Z,OPENGO
                ;
                ;REOPEN THEN CLOSE
EA24    D5              PUSH    DE
EA25    C5              PUSH    BC
EA26    CD EB84         CALL    CLOSE
EA29    C1              POP     BC
EA2A    D1              POP     DE
                ;
EA2B    3E FF   OPENGO: LD      A,0FFH
EA2D    32 EE60         LD      (OPENF),A       ;SET FLAG
                ;
EA30    3D      OPENW:  DEC     A
EA31    20 FD           JR      NZ,OPENW        ;WAIT
                ;
EA33    F3              DI
EA34    21 EBC6         LD      HL,INTRSR
EA37    22 005C         LD      (INTRSV),HL
EA3A    FB              EI              ;SET VECTOR
                ;
EA3B    EB              EX      DE,HL
EA3C    78              LD      A,B     ;COPY LEN
EA3D    B7              OR      A
EA3E    CA EE26         JP      Z,ERR   ;NULL STR
EA41    05              DEC     B
EA42    7E              LD      A,(HL)  ;1 CHAR
EA43    23              INC     HL
EA44    FE 49           CP      'I'
EA46    28 0B           JR      Z,OPENOK
EA48    FE 4F           CP      'O'
EA4A    28 07           JR      Z,OPENOK
EA4C    FE 43           CP      'C'
EA4E    28 03           JR      Z,OPENOK
EA50    C3 EE26         JP      ERR
                ;
EA53    32 EE4B OPENOK: LD      (IOC),A ;I,O or C
EA56    78              LD      A,B     ;GET COUNT
EA57    D6 04           SUB     4
EA59    DA EE26         JP      C,ERR
EA5C    FE 07           CP      7
EA5E    D2 EE26         JP      NC,ERR
EA61    4F              LD      C,A     ;COUNTER
EA62    7E              LD      A,(HL)
EA63    23              INC     HL
                ;
                ;BAU RATE       0-6
EA64    D6 30           SUB     '0'
EA66    FE 07           CP      7
EA68    D2 EE26         JP      NC,ERR
EA6B    16 D0           LD      D,208
EA6D    D6 02           SUB     2
EA6F    F5              PUSH    AF
EA70    30 01           JR      NC,OPEN0
EA72    AF              XOR     A
EA73    B7      OPEN0:  OR      A
EA74    28 05           JR      Z,OPEN1
EA76    CB 3A           SRL     D       ;D=D/2
EA78    3D              DEC     A
EA79    18 F8           JR      OPEN0
                ;
EA7B    7A      OPEN1:  LD      A,D
EA7C    32 EE2E         LD      (CTR),A
                ;
EA7F    F1              POP     AF
EA80    16 40           LD      D,40H           ;*16
EA82    30 07           JR      NC,OPEN2
EA84    16 80           LD      D,80H           ;*32
EA86    3C              INC     A
EA87    28 02           JR      Z,OPEN2
EA89    16 C0           LD      D,0C0H          ;*64
                ;
EA8B    5A      OPEN2:  LD      E,D             ;KEEP
                ;
                ;PARITY         E,O,N
EA8C    7E              LD      A,(HL)
EA8D    23              INC     HL
EA8E    16 00           LD      D,0
EA90    FE 4E           CP      'N'
EA92    28 0B           JR      Z,OPEN3
EA94    14              INC     D               ;D=1
EA95    FE 4F           CP      'O'
EA97    28 06           JR      Z,OPEN3
EA99    14              INC     D               ;D=2
EA9A    FE 45           CP      'E'
EA9C    C2 EE26         JP      NZ,ERR
EA9F    7B      OPEN3:  LD      A,E
EAA0    B2              OR      D
EAA1    5F              LD      E,A             ;KEEP
                ;
                ;DETA BIT LEN   5,6,7,8
EAA2    7E              LD      A,(HL)
EAA3    23              INC     HL
EAA4    D6 35           SUB     '5'
EAA6    DA EE26         JP      C,ERR
EAA9    FE 04           CP      4
EAAB    D2 EE26         JP      NC,ERR
                ;A=0-3
EAAE    F5              PUSH    AF
EAAF    16 E0           LD      D,11100000B
EAB1    B7              OR      A
EAB2    28 05           JR      Z,OPEN35
EAB4    CB 22   MLOOP:  SLA     D
EAB6    3D              DEC     A
EAB7    20 FB           JR      NZ,MLOOP
                ;
EAB9    7A      OPEN35: LD      A,D
EABA    2F              CPL
EABB    32 EE61         LD      (MASK),A
EABE    F1              POP     AF
EABF    B7              OR      A
EAC0    EA EAC5         JP      PE,OPEN4
EAC3    EE 03           XOR     11B
EAC5    0F      OPEN4:  RRCA
EAC6    0F              RRCA
EAC7    F6 01           OR      1
EAC9    32 EE5A         LD      (WR3+1),A       ;RCV
EACC    0F              RRCA
EACD    F6 80           OR      80H
EACF    F6 02           OR      02H
EAD1    F6 08           OR      08H
EAD3    32 EE54         LD      (WR5+1),A       ;SND
                ;
                ;STOP BIT LEN   1,2,3
EAD6    7E              LD      A,(HL)
EAD7    23              INC     HL
EAD8    D6 31           SUB     '1'
EADA    DA EE26         JP      C,ERR
EADD    FE 03           CP      3
EADF    D2 EE26         JP      NC,ERR
EAE2    3C              INC     A
EAE3    87              ADD     A,A
EAE4    87              ADD     A,A     ;A=A<<2
EAE5    B3              OR      E
EAE6    32 EE52         LD      (WR4+1),A
                ;
                ;CONTROLE       X,R,N,""
                ;KANA MODE      S,N,""
                ;CR,LF  SEND    C,L,""
                ;CR,LF  RECEIVE C,L,""
                ;KANJI          J,N,""  ;DUMMY
                ;END CODE       '@'-'_'
                ;
EAE9    C5              PUSH    BC
EAEA    E5              PUSH    HL
EAEB    21 EE3F         LD      HL,PDFLT
EAEE    11 EE45         LD      DE,PARAM
EAF1    01 0006         LD      BC,6
EAF4    ED B0           LDIR            ;COPY DEFAULT
EAF6    EB              EX      DE,HL   ;DE'=PARAM
EAF7    D9              EXX
EAF8    E1              POP     HL
EAF9    C1              POP     BC
EAFA    11 EE2F         LD      DE,PLIST
EAFD    AF              XOR     A       ;FLAG
EAFE    08              EX      AF,AF'  ;A'=0
EAFF    79              LD      A,C
EB00    B7              OR      A
EB01    28 1F           JR      Z,FINIS
EB03    FE 06           CP      6
EB05    20 02           JR      NZ,OPEN6
EB07    0D              DEC     C       ;C=5
EB08    08              EX      AF,AF'  ;A'<>0
                ;
EB09    46      OPEN6:  LD      B,(HL)
EB0A    23              INC     HL
EB0B    1A      OPEN7:  LD      A,(DE)
EB0C    13              INC     DE
EB0D    B7              OR      A       ;SEPARATOR
EB0E    CA EE26         JP      Z,ERR
EB11    B8              CP      B
EB12    20 F7           JR      NZ,OPEN7
                ;MATCH
EB14    D9              EXX
EB15    12              LD      (DE),A
EB16    D9              EXX
EB17    1A      OPEN75: LD      A,(DE)
EB18    13              INC     DE
EB19    B7              OR      A
EB1A    20 FB           JR      NZ,OPEN75
EB1C    D9      OPEN8:  EXX
EB1D    13              INC     DE
EB1E    D9              EXX
EB1F    0D              DEC     C
EB20    20 E7           JR      NZ,OPEN6
                ;
EB22    08      FINIS:  EX      AF,AF'
EB23    B7              OR      A
EB24    28 10           JR      Z,SETZ
EB26    7E              LD      A,(HL)  ;ENDCODE
EB27    FE 40           CP      40H
EB29    DA EE26         JP      C,ERR
EB2C    FE 80           CP      80H
EB2E    D2 EE26         JP      NC,ERR
EB31    E6 1F           AND     1FH
EB33    D9              EXX
EB34    12              LD      (DE),A  ;STORE
EB35    D9              EXX
                ;
EB36    F3      SETZ:   DI              ;SET ZCTC,ZSIO
```

```
EB37    CD EB6A              CALL    INITBF
EB3A    01 1FA9              LD      BC,ZCTC+1        ;CH0
EB3D    3E 47                LD      A,01000111B      ;MODE
EB3F    ED 79                OUT     (C),A
EB41    3A EE2E              LD      A,(CTR)
EB44    ED 79                OUT     (C),A            ;T CONSTANT
                     ;
EB46    21 EE4C              LD      HL,SIOSA
EB49    01 1F99              LD      BC,ZSIO+1        ;CH A
EB4C    3E 0F                LD      A,15
EB4E    CD EB63              CALL    SETSIO
                     ;
EB51    21 EE5B              LD      HL,SIOSB
EB54    01 1F9B              LD      BC,ZSIO+3        ;CH B
EB57    3E 05                LD      A,5
EB59    CD EB63              CALL    SETSIO
EB5C    3E FF                LD      A,0FFH
EB5E    32 EE60              LD      (OPENF),A
EB61    FB                   EI
EB62    C9                   RET
                     ;
EB63    04           SETSIO: INC     B
EB64    ED A3                OUTI
EB66    3D                   DEC     A
EB67    20 FA                JR      NZ,SETSIO
EB69    C9                   RET
                     ;
EB6A    AF           INITBF: XOR     A
EB6B    32 EE6A              LD      (QLEN),A
EB6E    21 EE6B              LD      HL,Q0
EB71    22 EE66              LD      (HEAD),HL
EB74    22 EE68              LD      (TAIL),HL
                     ;
EB77    32 EE62              LD      (LASTX),A
EB7A    32 EE63              LD      (XFLAG),A
EB7D    32 EE64              LD      (TKIN),A
EB80    32 EE65              LD      (RKIN),A         ;RESET MODE
EB83    C9                   RET
                     ;
EB84    AF           CLOSE:  XOR     A
EB85    32 EE60              LD      (OPENF),A
                     ;
EB88    3A EE4B              LD      A,(IOC)
EB8B    FE 4F                CP      'O'
EB8D    C0                   RET     NZ
EB8E    3A EE4A              LD      A,(ECODE)
EB91    FE 20                CP      20H
EB93    C8                   RET     Z
EB94    CD EDCD              CALL    PPUTC   ;TX ECODE
EB97    C9                   RET
                     ;
EB98    3A EE6A      LOC:    LD      A,(QLEN)
EB9B    77                   LD      (HL),A
EB9C    23                   INC     HL
EB9D    36 00                LD      (HL),0  ;RET SIZE
EB9F    C9                   RET
                     ;
                     ;BREAK B,A,F
EBA0    3A EE6A      QEOF:   LD      A,(QLEN)
EBA3    B7                   OR      A
EBA4    C8                   RET     Z       ;EMPTY
EBA5    3A EE4A              LD      A,(ECODE)
EBA8    FE 20                CP      20H
EBAA    C8                   RET     Z       ;NO ECODE
EBAB    E5                   PUSH    HL
EBAC    2A EE68              LD      HL,(TAIL)
EBAF    46                   LD      B,(HL)  ;NOZOKU
EBB0    E1                   POP     HL
EBB1    B8                   CP      B
EBB2    28 02                JR      Z,QEOF1
EBB4    AF                   XOR     A       ;SET Z
EBB5    C9                   RET
EBB6    F6 FF        QEOF1:  OR      0FFH    ;SET NZ
EBB8    C9                   RET
                     ;
EBB9    CD EBA0      EOF:    CALL    QEOF
EBBC    11 0000              LD      DE,0
EBBF    28 01                JR      Z,EOF1
EBC1    1B                   DEC     DE
EBC2    73           EOF1:   LD      (HL),E
EBC3    23                   INC     HL
EBC4    72                   LD      (HL),D
EBC5    C9                   RET
                     ;
EBC6    F3           INTRSR: DI                      ;INT ROUTINE
EBC7    F5                   PUSH    AF
EBC8    C5                   PUSH    BC
EBC9    D5                   PUSH    DE
EBCA    E5                   PUSH    HL
                     ;
EBCB    01 1F99              LD      BC,ZSIO+1
                     ;
EBCE    3E 01        INTR1:  LD      A,1
EBD0    ED 79                OUT     (C),A   ;RR1
EBD2    ED 78                IN      A,(C)
                     ;
EBD4    0B                   DEC     BC
EBD5    ED 78                IN      A,(C)
EBD7    2A EE66              LD      HL,(HEAD)
EBDA    77                   LD      (HL),A
EBDB    23                   INC     HL
EBDC    E5                   PUSH    HL
EBDD    01 EEEB              LD      BC,Q1
EBE0    B7                   OR      A
EBE1    ED 42                SBC     HL,BC
EBE3    E1                   POP     HL
EBE4    20 03                JR      NZ,INTR2
EBE6    21 EE6B              LD      HL,Q0            ;QUE !
EBE9    22 EE66      INTR2:  LD      (HEAD),HL
EBEC    21 EE6A              LD      HL,QLEN
EBEF    34                   INC     (HL)
                     ;
EBF0    FE 13                CP      XOFF
EBF2    20 05                JR      NZ,INTR3
EBF4    32 EE62              LD      (LASTX),A        ;GET XOFF
EBF7    18 08                JR      INTR4
EBF9    FE 11        INTR3:  CP      XON
EBFB    20 04                JR      NZ,INTR4
EBFD    AF                   XOR     A
EBFE    32 EE62              LD      (LASTX),A        ;GET XON
EC01    3A EE6A      INTR4:  LD      A,(QLEN)
EC04    3C                   INC     A
EC05    FE 40                CP      QMAX
EC07    D4 EC11              CALL    NC,SWAIT         ;SEND WAIT
                     ;
EC0A    E1                   POP     HL
EC0B    D1                   POP     DE
EC0C    C1                   POP     BC
EC0D    F1                   POP     AF
EC0E    FB                   EI
EC0F    ED 4D                RETI
                     ;
```

```
EC11    01 1F99      SWAIT:  LD      BC,ZSIO+1
EC14    3A EE45              LD      A,(PARAM+0)
EC17    FE 4E                CP      'N'
EC19    C8                   RET     Z       ;NOTHING
EC1A    FE 52                CP      'R'
EC1C    28 11                JR      Z,SWAIT1
EC1E    FE 58                CP      'X'
EC20    C0                   RET     NZ
EC21    3A EE63              LD      A,(XFLAG)
EC24    B7                   OR      A
EC25    C0                   RET     NZ
EC26    3E 13                LD      A,XOFF  ;X CTRL
EC28    32 EE63              LD      (XFLAG),A
EC2B    CD ED77              CALL    PPUTC0
EC2E    C9                   RET
EC2F    01 1F99      SWAIT1: LD      BC,ZSIO+1
EC32    3E 05                LD      A,5
EC34    ED 79                OUT     (C),A
EC36    3A EE54              LD      A,(WR5+1)
EC39    E6 FD                AND     11111101B        ;RES RTS
EC3B    ED 79                OUT     (C),A
EC3D    C9                   RET
                     ;
EC3E    3A EE45      SOK:    LD      A,(PARAM+0)
EC41    FE 4E                CP      'N'
EC43    C8                   RET     Z
                     ;
EC44    FE 52                CP      'R'
EC46    28 12                JR      Z,SOK1
EC48    FE 58                CP      'X'
EC4A    C0                   RET     NZ
EC4B    3A EE63              LD      A,(XFLAG)
EC4E    B7                   OR      A
EC4F    C8                   RET     Z
EC50    AF                   XOR     A
EC51    32 EE63              LD      (XFLAG),A
EC54    3E 11                LD      A,XON
EC56    CD ED77              CALL    PPUTC0
EC59    C9                   RET
EC5A    01 1F99      SOK1:   LD      BC,ZSIO+1
EC5D    3E 05                LD      A,5
EC5F    ED 79                OUT     (C),A
EC61    3A EE54              LD      A,(WR5+1)
EC64    F6 02                OR      00000010B        ;SET RTS
EC66    ED 79                OUT     (C),A
EC68    C9                   RET
                     ;
EC69    CD ECA4      XGETC:  CALL    PGETC   ;GET CHAR
EC6C    47                   LD      B,A     ;SAVE
EC6D    3A EE46              LD      A,(PARAM+1)
EC70    FE 53                CP      'S'
EC72    20 24                JR      NZ,XGETC4
EC74    3A EE61              LD      A,(MASK)
EC77    FE 7F                CP      1111111B
EC79    20 1D                JR      NZ,XGETC4
                     ;
                     ;S AND 7 BITS
EC7B    3A EE65              LD      A,(RKIN)
EC7E    B7                   OR      A
EC7F    78                   LD      A,B
EC80    20 06                JR      NZ,XGETC1
EC82    FE 0E                CP      SO      ;TO KANA MODE
EC84    28 07                JR      Z,XGETC2
EC86    18 10                JR      XGETC4
                     ;
EC88    FE 0F        XGETC1: CP      SI
EC8A    20 06                JR      NZ,XGETC3
EC8C    AF                   XOR     A
EC8D    32 EE65      XGETC2: LD      (RKIN),A         ;CLEAR
EC90    18 D7                JR      XGETC            ;AGAIN
                     ;
EC92    FE 20        XGETC3: CP      0A0H-080H
EC94    38 02                JR      C,XGETC4
EC96    CB F8                SET     7,B     ;TO KANA
EC98    78           XGETC4: LD      A,B
EC99    C9                   RET
                     ;
EC9A    E5           PGETC1: PUSH    HL      ;RETURN CHAR
EC9B    CD ECA4              CALL    PGETC
EC9E    E1                   POP     HL
EC9F    77                   LD      (HL),A
ECA0    23                   INC     HL
ECA1    36 00                LD      (HL),0
ECA3    C9                   RET
                     ;
ECA4    CD 0330      PGETC:  CALL    QBREAK
ECA7    CA EE2A              JP      Z,BREAK ;BREAK?
ECAA    3A EE6A              LD      A,(QLEN)
ECAD    B7                   OR      A
ECAE    28 F4                JR      Z,PGETC
                     ;
ECB0    F3                   DI
ECB1    3A EE6A              LD      A,(QLEN)         ;RE-GET
ECB4    5F                   LD      E,A              ;SAVE QLEN
ECB5    2A EE68              LD      HL,(TAIL)
ECB8    3A EE61              LD      A,(MASK)
ECBB    A6                   AND     (HL)
ECBC    57                   LD      D,A     ;GET DATA
ECBD    23                   INC     HL
ECBE    E5                   PUSH    HL
ECBF    01 EEEB              LD      BC,Q1
ECC2    B7                   OR      A
ECC3    ED 42                SBC     HL,BC
ECC5    E1                   POP     HL
ECC6    20 03                JR      NZ,GETC1
ECC8    21 EE6B              LD      HL,Q0            ;QUE !
ECCB    22 EE68      GETC1:  LD      (TAIL),HL
ECCE    7B                   LD      A,E              ;GET QLEN
ECCF    3D                   DEC     A
ECD0    32 EE6A              LD      (QLEN),A
ECD3    FE 30                CP      QMAX-16
ECD5    D5                   PUSH    DE
ECD6    DC EC3E              CALL    C,SOK            ;SEND OK
ECD9    FB                   EI
ECDA    D1                   POP     DE
ECDB    7A                   LD      A,D
ECDC    C9                   RET
                     ;
ECDD    EB           LINPUT: EX      DE,HL
ECDE    E5                   PUSH    HL
ECDF    23                   INC     HL
ECE0    78                   LD      A,B
ECE1    FE 02                CP      2
ECE3    DA EE26              JP      C,ERR   ;NO SPACE
ECE6    05                   DEC     B       ;DEC COUNTER
ECE7    0E 00                LD      C,0
                     ;
ECE9    E5           LIP1:   PUSH    HL
ECEA    C5                   PUSH    BC      ;SAVE COUNTER
ECEB    CD 0330      LIP1L:  CALL    QBREAK  ;BREAK?
ECEE    CA EE2A              JP      Z,BREAK
ECF1    3A EE6A              LD      A,(QLEN)
```

▶今回は，主人に代わり私が書きます。新婚時代はよくマシン語の入力（本の記事から）をしました。最近ではスロットマシンゲームをOh!MZから¾ほど入力してまして，まだ完成していません。早く走らせたいと思います。　岡本　園子（26）愛知県

```
ECF4    B7                      OR      A
ECF5    28 F4                   JR      Z,LIP1L
ECF7    CD EBA0                 CALL    QEOF
ECFA    28 0A                   JR      Z,LIP1L1
                        ;GET EOF
ECFC    C1                      POP     BC
ECFD    E1                      POP     HL
ECFE    79                      LD      A,C
ECFF    FE 0D                   CP      CR
ED01    20 2A                   JR      NZ,LIPL2
ED03    2B                      DEC     HL
ED04    18 27                   JR      LIPL2   ;SUTE CR
                        ;
ED06    CD EC69         LIP1L1: CALL    XGETC
ED09    C1                      POP     BC
ED0A    E1                      POP     HL
                        ;1 LINE?
ED0B    F5                      PUSH    AF
ED0C    3A EE48                 LD      A,(PARAM+3)
ED0F    FE 4C                   CP      'L'
ED11    28 07                   JR      Z,LIPL1
ED13    F1                      POP     AF
ED14    FE 0D                   CP      CR
ED16    28 15                   JR      Z,LIPL2
ED18    18 0E                   JR      LIP2
ED1A    79              LIPL1:  LD      A,C
ED1B    FE 0D                   CP      CR
ED1D    20 08                   JR      NZ,LIPL3
ED1F    F1                      POP     AF
ED20    FE 0A                   CP      LF
ED22    20 04                   JR      NZ,LIP2
ED24    2B                      DEC     HL
ED25    18 06                   JR      LIPL2
                        ;
ED27    F1              LIPL3:  POP     AF
ED28    77              LIP2:   LD      (HL),A
ED29    4E              LIP3:   LD      C,(HL)  ;LAST CHAR
ED2A    23                      INC     HL
ED2B    10 BC                   DJNZ    LIP1    ;FULL?
ED2D    D1              LIPL2:  POP     DE
ED2E    B7                      OR      A
ED2F    ED 52                   SBC     HL,DE
ED31    EB                      EX      DE,HL
ED32    1D                      DEC     E
ED33    73                      LD      (HL),E  ;STORE LEN
ED34    C9                      RET
                        ;
                        ;INPUT n CHAR
ED35    D5              INPUTN: PUSH    DE      ;SAVE ADDR
ED36    EB                      EX      DE,HL
ED37    7E                      LD      A,(HL)
ED38    23                      INC     HL
ED39    B8                      CP      B
ED3A    D2 EE26                 JP      NC,ERR  ;TOO SHORT
                        ;A<B
ED3D    05                      DEC     B
ED3E    B7                      OR      A
ED3F    28 12                   JR      Z,IPN2  ;REQUEST 0 CHAR
ED41    47                      LD      B,A     ;COUNTER
ED42    0E 00                   LD      C,0     ;LEN
                        ;
ED44    C5              IPN1:   PUSH    BC
ED45    E5                      PUSH    HL
ED46    CD EC69                 CALL    XGETC
ED49    E1                      POP     HL
ED4A    C1                      POP     BC
ED4B    77                      LD      (HL),A  ;STORE CODE
ED4C    23                      INC     HL
ED4D    0C                      INC     C
ED4E    10 F4                   DJNZ    IPN1
ED50    E1                      POP     HL      ;TOP ADDR.
ED51    71                      LD      (HL),C  ;STORE LEN
ED52    C9                      RET
                        ;
ED53    D1              IPN2:   POP     DE
ED54    12                      LD      (DE),A  ;STORE 0
ED55    C9                      RET
                        ;
ED56    CD ED68         PRINT:  CALL    PRINTS
ED59    3E 0D                   LD      A,CR
ED5B    CD EDCD                 CALL    PPUTC
ED5E    3A EE47                 LD      A,(PARAM+2)
ED61    FE 4C                   CP      'L'
ED63    C0                      RET     NZ      ;'C'
ED64    3E 0A                   LD      A,LF    ;'L'
ED66    18 65                   JR      PPUTC
                        ;
ED68    78              PRINTS: LD      A,B
ED69    B7                      OR      A       ;CK LEN
ED6A    C8                      RET     Z
ED6B    1A              PUTSL:  LD      A,(DE)
ED6C    13                      INC     DE
ED6D    C5                      PUSH    BC
ED6E    D5                      PUSH    DE
ED6F    CD ED93                 CALL    PUTC
ED72    D1                      POP     DE
ED73    C1                      POP     BC
ED74    10 F5                   DJNZ    PUTSL
ED76    C9                      RET
                        ;
ED77    F5              PPUTC0: PUSH    AF
ED78    01 1F99                 LD      BC,ZSIO+1
ED7B    3E 10           PUTCL0: LD      A,10H   ;RESET STAT
ED7D    ED 79                   OUT     (C),A
ED7F    ED 78                   IN      A,(C)   ;GET RR0
ED81    CB 57                   BIT     2,A     ;TX BUFF
ED83    28 F6                   JR      Z,PUTCL0
                        ;
ED85    CD EE09                 CALL    CHTXOK  ;TX OK?
ED88    28 62                   JR      Z,PUTOK
ED8A    3E 10                   LD      A,10H   ;STAT RESET
ED8C    ED 79                   OUT     (C),A
ED8E    18 EB                   JR      PUTCL0  ;LOOP
                        ;
ED90    7E              PPUTC1: LD      A,(HL)
ED91    18 3A                   JR      PPUTC
                        ;
ED93    47              PUTC:   LD      B,A
ED94    3A EE46                 LD      A,(PARAM+1)
ED97    FE 53                   CP      'S'
ED99    20 31                   JR      NZ,PUTC0
                        ;
ED9B    3A EE61                 LD      A,(MASK)
ED9E    FE 7F                   CP      1111111B
EDA0    20 2A                   JR      NZ,PUTC0
                        ;
                        ;'S' AND 7 BITS
EDA2    CB 78                   BIT     7,B
EDA4    28 15                   JR      Z,NKANA ;NOT KANA
EDA6    3A EE64                 LD      A,(TKIN)
EDA9    B7                      OR      A
EDAA    20 0A                   JR      NZ,IKANA
EDAC    C5                      PUSH    BC      ;SAVE IT
EDAD    3E 0F                   LD      A,SI    ;SHIFT IN
EDAF    CD EDCD                 CALL    PPUTC
EDB2    32 EE64                 LD      (TKIN),A
EDB5    C1                      POP     BC
                        ;
EDB6    78              IKANA:  LD      A,B
EDB7    E6 7F                   AND     7FH     ;SEND 7 BITS
EDB9    18 12                   JR      PPUTC
                        ;
EDBB    3A EE64         NKANA:  LD      A,(TKIN)
EDBE    B7                      OR      A
EDBF    28 0B                   JR      Z,PUTC0 ;NO KANA MODE
                        ;
EDC1    C5                      PUSH    BC      ;SAVE IT
EDC2    3E 0E                   LD      A,SO    ;SHIFT OUT
EDC4    CD EDCD                 CALL    PPUTC   ;PUT ANY HOW
EDC7    AF                      XOR     A
EDC8    32 EE64                 LD      (TKIN),A
EDCB    C1                      POP     BC      ;B=IT
                        ;
EDCC    78              PUTC0:  LD      A,B     ;DATA TO TX
EDCD    F5              PPUTC:  PUSH    AF
EDCE    01 1F99                 LD      BC,ZSIO+1
EDD1    CD 0330         PUTCL:  CALL    QBREAK  ;BREAK?
EDD4    CA EE2A                 JP      Z,BREAK
EDD7    3E 10                   LD      A,10H   ;RESET STAT
EDD9    ED 79                   OUT     (C),A
EDDB    ED 78                   IN      A,(C)   ;GET RR0
EDDD    CB 57                   BIT     2,A     ;TX BUFF
EDDF    28 F0                   JR      Z,PUTCL
                        ;
EDE1    CD EE09                 CALL    CHTXOK  ;TX OK?
EDE4    28 06                   JR      Z,PUTOK
EDE6    3E 10                   LD      A,10H   ;STAT RESET
EDE8    ED 79                   OUT     (C),A
EDEA    18 E5                   JR      PUTCL   ;LOOP
                        ;
EDEC    0B              PUTOK:  DEC     BC      ;DATA PORT
EDED    D1                      POP     DE      ;D=CHAR
EDEE    3A EE61                 LD      A,(MASK)
EDF1    A2                      AND     D
EDF2    ED 79                   OUT     (C),A   ;TX DATA
                        ;
EDF4    3A EE6A                 LD      A,(QLEN)
EDF7    B7                      OR      A       ;EMPTY?
EDF8    C8                      RET     Z
                        ;
                        ;DROP XON,XOFF
EDF9    2A EE68         PEND0:  LD      HL,(TAIL)
EDFC    7E                      LD      A,(HL)  ;NOZOKU
EDFD    FE 11                   CP      XON
EDFF    28 03                   JR      Z,PEND1
EE01    FE 13                   CP      XOFF
EE03    C0                      RET     NZ
EE04    CD EC9A         PEND1:  CALL    PGETC1  ;DROP
EE07    18 F0                   JR      PEND0
                        ;
EE09    57              CHTXOK: LD      D,A     ;SAVE RR0
EE0A    3A EE45                 LD      A,(PARAM+0)
EE0D    FE 4E                   CP      'N'     ;NO CON
EE0F    C8                      RET     Z
EE10    FE 52                   CP      'R'     ;RTS
EE12    20 05                   JR      NZ,CHX
EE14    7A                      LD      A,D
EE15    2F                      CPL             ;REVERSE
EE16    CB 6F                   BIT     5,A     ;RTS
EE18    C9                      RET
                        ;
EE19    FE 58           CHX:    CP      'X'
EE1B    28 02                   JR      Z,CHX1
EE1D    AF                      XOR     A
EE1E    C9                      RET
EE1F    F3              CHX1:   DI              ;ENGIMON
EE20    3A EE62                 LD      A,(LASTX)
EE23    B7                      OR      A
EE24    FB                      EI
EE25    C9                      RET
                        ;
EE26    3E 05           ERR:    LD      A,5     ;ILL FUNC
EE28    DD E9                   JP      (IX)
                        ;
EE2A    F3              BREAK:  DI
EE2B    C3 1FCA                 JP      BBREAK
                        ;
EE2E                    CTR:    DS      1       ;CTC COUNTER
                        ;
EE2F    58 52 4E 00     PLIST:  DB      'XRN',0
EE33    53 4E 00                DB      'SN',0
EE36    43 4C 00                DB      'CL',0
EE39    43 4C 00                DB      'CL',0
EE3C    4A 4E 00                DB      'JN',0
                        ;
EE3F    4E 4E 4C 4C     PDFLT:  DB      'NNLLN',20H
EE43    4E 20
                        ;
EE45                    PARAM:  DS      5       ;P AREA
EE4A                    ECODE:  DS      1       ;5+1=6
                        ;
EE4B                    IOC:    DS      1       ;OPEN MODE FLAG
                        ;
EE4C    18              SIOSA:  DB      00011000B       ;WR0
EE4D    01 10           WR1:    DB      1,10H
EE4F    02 00           WR2:    DB      2,00H
EE51    04 00           WR4:    DB      4,00H
EE53    05 00           WR5:    DB      5,00H
EE55    06 00           WR6:    DB      6,00H
EE57    07 00           WR7:    DB      7,00H
EE59    03 00           WR3:    DB      3,00H
                        ;
EE5B    18              SIOSB:  DB      18H
EE5C    01 00                   DB      1,00H
EE5E    02 5C                   DB      2,INTRSV
                        ;
EE60    00              OPENF:  DB      0       ;OPEB FLAG
                        ;
EE61                    MASK:   DS      1       ;DATA MASK
EE62                    LASTX:  DS      1       ;R FLAG
EE63                    XFLAG:  DS      1       ;T FLAG
                        ;
EE64                    TKIN:   DS      1       ;R KANA FLAG
EE65                    RKIN:   DS      1       ;T KANA FLAG
                        ;
EE66                    HEAD:   DS      2       ;QUE HEAD
EE68                    TAIL:   DS      2       ;QUE TAIL
EE6A                    QLEN:   DS      1       ;Q LENGTH
EE6B                    Q0:     DS      QMAXX
EEEB                    Q1:
                        ;
                                END
```

▶富士通の9450へIBMのメインコンピュータから送られてくるデータを，X1 turbo Zで読み出すことができました。Oh!MZ1986年2月号の8/5インチコンバータのEBCDIC→ASC変換を使わせてもらったのです。ED5B番地はØØH→2ØHにしました。

宮城　照彦（40）千葉県

リスト2 ダンプリスト

```
EA00 C3 1E EA C3 84 EB C3 DD : 9D
EA08 EC C3 35 ED C3 68 ED C3 : AC
EA10 56 ED C3 98 EB C3 B9 EB : F0
EA18 C3 9A EC C3 90 ED 3A 60 : 23
EA20 EE B7 28 07 D5 C5 CD 84 : BF
EA28 EB C1 D1 3E FF 32 60 EE : 3A
EA30 3D 20 FD F3 21 C6 EB 22 : 41
EA38 5C 00 FB EB 78 B7 CA 26 : 61
EA40 EE 05 7E 23 FE 49 28 0B : 0E
EA48 FE 4F 28 07 FE 43 28 03 : E8
EA50 C3 26 EE 32 4B EE 78 D6 : 90
EA58 04 DA 26 EE FE 07 D2 26 : EF
EA60 EE 4F 7E 23 D6 30 FE 07 : E9
EA68 D2 26 EE 16 D0 D6 02 F5 : 99
EA70 30 01 AF B7 28 05 CB 3A : C9
EA78 3D 18 F8 7A 32 2E EE F1 : 06
---------------------------------
SUM: 1A E2 8C E2 74 31 D8 D6 6C5A

EA80 16 40 30 07 16 80 3C 28 : 87
EA88 02 16 C0 5A 7E 23 16 00 : E9
EA90 FE 4E 28 0B 14 FE 4F 28 : 08
EA98 06 14 FE 45 C2 26 EE 7B : AE
EAA0 B2 5F 7E 23 D6 35 DA 26 : BD
EAA8 EE FE 04 D2 26 EE F5 16 : E1
EAB0 E0 B7 28 05 CB 22 3D 20 : 0E
EAB8 FB 7A 2F 32 61 EE F1 B7 : CD
EAC0 EA C5 EA EE 03 0F 0F F6 : 9E
EAC8 01 32 5A EE 0F F6 80 F6 : F6
EAD0 02 F6 08 32 54 EE 7E 23 : 15
EAD8 D6 31 DA 26 EE FE 03 D2 : C8
EAE0 26 EE 3C 87 87 B3 32 52 : 95
EAE8 EE C5 E5 21 3F EE 11 45 : 3C
EAF0 EE 01 06 00 ED B0 EB D9 : 56
EAF8 E1 C1 11 2F EE AF 08 79 : 00
---------------------------------
SUM: 3D D9 4D E8 87 EB D2 A8 CC3B

EB00 B7 28 1F FE 06 20 02 0D : 31
EB08 08 46 23 1A 13 B7 CA 26 : 45
EB10 EE B8 20 F7 D9 12 D9 1A : 9B
EB18 13 B7 20 FB D9 13 D9 0D : B7
EB20 20 E7 08 B7 28 10 7E FE : 7A
EB28 40 DA 26 EE FE 80 D2 26 : A4
EB30 EE E6 1F D9 12 D9 F3 CD : 77
EB38 6A EB 01 A9 1F 3E 47 ED : 90
EB40 79 3A 2E EE ED 79 21 4C : A2
EB48 EE 01 99 1F 3E 0F CD 63 : 24
EB50 EB 21 5B EE 01 9B 1F 3E : 4E
EB58 05 CD 63 EB 3E FF 32 60 : EF
EB60 EE FB C9 04 ED A3 3D 20 : A3
EB68 FA C9 AF 32 6A EE 21 6B : 88
EB70 EE 22 66 EE 22 68 EE 32 : 0E
EB78 62 EE 32 63 EE 32 64 EE : 57
---------------------------------
SUM: 07 6C 65 9E F3 F0 F7 30 1771

EB80 32 65 EE C9 AF 32 60 EE : 7D
EB88 3A 4B EE FE 4F C0 3A 4A : 04
EB90 EE FE 20 C8 CD CD ED C9 : 24
EB98 3A 6A EE 77 23 36 00 C9 : 2B
EBA0 3A 6A EE B7 C8 3A 4A EE : 83
EBA8 FE 20 C8 E5 2A 68 EE 46 : 91
EBB0 E1 B8 28 02 AF C9 F6 FF : 30
EBB8 C9 CD A0 EB 11 00 00 28 : 5A
EBC0 01 1B 73 23 72 C9 F3 F5 : D5
EBC8 C5 D5 E5 01 99 1F 3E 01 : 77
EBD0 ED 79 ED 78 0B ED 78 2A : 65
EBD8 66 EE 77 23 E5 01 EB EE : AD
EBE0 B7 ED 42 E1 20 03 21 6B : 76
EBE8 EE 22 66 EE 21 6A EE 34 : 11
EBF0 FE 13 20 05 32 62 EE 18 : D0
EBF8 08 FE 11 20 04 AF 32 62 : 7E
---------------------------------
SUM: 3A 9E FD 42 12 B4 78 4C 99CD

EC00 EE 3A 6A EE 3C FE 40 D4 : CE
EC08 11 EC E1 D1 C1 F1 FB ED : 49
EC10 4D 01 99 1F 3A 45 EE FE : 71
EC18 4E C8 FE 52 28 11 FE 58 : F5
EC20 C0 3A 63 EE B7 C0 3E 13 : 13
EC28 32 63 EE CD 77 ED C9 01 : 7E
EC30 99 1F 3E 05 ED 79 3A 54 : EF
EC38 EE E6 FD ED 79 C9 3A 45 : 7F
EC40 EE FE 4E C8 FE 52 28 12 : 8C
EC48 FE 58 C0 3A 63 EE B7 C8 : 20
EC50 AF 32 63 EE 3E 11 CD 77 : C5
EC58 ED C9 01 99 1F 3E 05 ED : 9F
EC60 79 3A 54 EE F6 02 ED 79 : 53
EC68 C9 CD A4 EC 47 3A 46 EE : DB
EC70 FE 53 20 24 3A 61 EE FE : 1C
EC78 7F 20 1D 3A 65 EE B7 78 : 78
---------------------------------
SUM: 5A 5C 15 9E 8D 4E 2B DF 9564

EC80 20 06 FE 0E 28 07 18 10 : 89
EC88 FE 0F 20 06 AF 32 65 EE : 67
EC90 18 D7 FE 20 38 02 CB F8 : 0A
EC98 78 C9 E5 CD A4 EC E1 77 : DB
ECA0 23 36 00 C9 CD 30 03 CA : EC
ECA8 2A EE 3A 6A EE B7 28 F4 : 7D
ECB0 F3 3A 6A EE 5F 2A 68 EE : 64
ECB8 3A 61 EE A6 57 23 E5 01 : 8F
ECC0 EB EE B7 ED 42 E1 20 03 : C3
ECC8 21 6B EE 22 68 EE 7B 3D : AA
ECD0 32 6A EE FE 30 D5 DC 3E : A7
ECD8 EC FB D1 7A C9 EB E5 23 : EE
ECE0 78 FE 02 DA 26 EE 05 0E : 79
ECE8 00 E5 C5 CD 30 03 CA 2A : 9E
ECF0 EE 3A 6A EE B7 28 F4 CD : 20
ECF8 A0 EB 28 0A C1 E1 79 FE : D6
---------------------------------
SUM: 58 3A 50 EE 95 E4 39 BE CDF1

ED00 0D 20 2A 2B 18 27 CD 69 : F7
ED08 EC C1 E1 F5 3A 48 EE FE : F1
ED10 4C 28 07 F1 FE 0D 28 15 : B4
ED18 18 0E 79 FE 0D 20 08 F1 : C3
ED20 FE 0A 20 04 2B 18 06 F1 : 66
ED28 77 4E 23 10 BC D1 B7 ED : 29
ED30 52 EB 1D 73 C9 D5 EB 7E : D4
ED38 23 B8 D2 26 EE 05 B7 28 : A5
ED40 12 47 0E 00 C5 E5 CD 69 : 47
ED48 EC E1 C1 77 23 0C 10 F4 : 38
ED50 E1 71 C9 D1 12 C9 CD 68 : FC
ED58 ED 3E 0D CD CD ED 3A 47 : 40
ED60 EE FE 4C C0 3E 0A 18 65 : BD
ED68 78 B7 C8 1A 13 C5 D5 CD : 8B
ED70 93 ED D1 C1 10 F5 C9 F5 : D5
ED78 01 99 1F 3E 10 ED 79 ED : 5A
---------------------------------
SUM: 0D 24 66 AA 33 B7 5D 11 8D3D

ED80 78 CB 57 28 F6 CD 09 EE : 7C
ED88 28 62 3E 10 ED 79 18 EB : 41
ED90 7E 18 3A 47 3A 46 EE FE : 83
ED98 53 20 31 3A 61 EE FE 7F : AA
EDA0 20 2A CB 78 28 15 3A 64 : 68
EDA8 EE B7 20 0A C5 3E 0F CD : AE
EDB0 CD ED 32 64 EE C1 78 E6 : 5D
EDB8 7F 18 12 3A 64 EE B7 28 : 14
EDC0 0B C5 3E 0E CD CD ED AF : 52
EDC8 32 64 EE C1 78 F5 01 99 : 4C
EDD0 1F CD 30 03 CA 2A EE 3E : 3F
EDD8 10 ED 79 ED 78 CB 57 28 : 25
EDE0 F0 CD 09 EE 28 06 3E 10 : 30
EDE8 ED 79 18 E5 0B D1 3A 61 : DA
EDF0 EE A2 ED 79 3A 6A EE B7 : 3F
EDF8 C8 2A 68 EE 7E FE 11 28 : FD
---------------------------------
SUM: CA 40 7A D2 2F 72 2F 93 6DF0

EE00 03 FE 13 C0 CD 9A EC 18 : 3F
EE08 F0 57 3A 45 EE FE 4E C8 : C8
EE10 FE 52 20 05 7A 2F CB 6F : 58
EE18 C9 FE 58 28 02 AF C9 F3 : B4
EE20 3A 62 EE B7 FB C9 3E 05 : 48
EE28 DD E9 F3 C3 CA 1F 15 58 : D2
EE30 52 4E 00 53 4E 00 43 4C : D0
EE38 00 43 4C 00 4A 4E 00 4E : 75
EE40 4E 4C 4C 4E 20 CD 83 21 : C5
EE48 47 C3 B3 20 18 01 10 02 : 08
EE50 00 04 00 05 00 06 00 07 : 16
EE58 00 03 00 18 01 00 02 5C : 7A
EE60 00                      : 00
---------------------------------
SUM: B8 97 F1 8A CD 80 F9 BF E0FD
```

リスト3 サンプルプログラム

```
100 CLEAR &HEA00
110 IF MEM$(&HEA00,3)<>HEXCHR$("C3 1E EA") LOADM "RS.OBJ"
120 DEFUSR0=&HEA00        :'OPEN    :STR
130 DEFUSR1=&HEA00+3      :'CLOSE   : *
140 DEFUSR2=&HEA00+6      :'LINPUT  :STR
150 DEFUSR3=&HEA00+9      :'INPUTN  :STR
160 DEFUSR4=&HEA00+12     :'PRINTS  :STR
170 DEFUSR5=&HEA00+15     :'PRINT   :STR
180 DEFUSR6=&HEA00+18     :'LOC     :INT
190 DEFUSR7=&HEA00+21     :'EOF     :INT
200 DEFUSR8=&HEA00+24     :'PGETC1  :INT
210 DEFUSR9=&HEA00+27     :'PPUTC1  :INT
220 '
230 A$=USR0("O6N83XNCCNZ")        :'OPEN
240 A$=USR1("")                   :'CLOSE
250 C=10:A$=USR2(STRING$(C,".")):'LINPUT
260 C=10:A$=USR3(CHR$(C)+STRING$(C,".")):'INPUTN
270 A$=USR4("STRINGS")            :'SEND STRING
280 A$=USR5("ONE LINE")           :'SEND 1 LINE
290 A=USR6(0)                     :'GET BUF SIZE
300 A=USR7(0)                     :'EOF ?
310 A=USR8(0)                     :'GET 1 BYTE
320 A=USR9(0)                     :'PUT 1 BYTE
```

リスト4 簡単なターミナルプログラム

```
220 '100 - 210 ｷﾞｮｳ ﾊ ﾘｽﾄ3 ﾉ ﾓﾉｦ ﾂｶｳﾉﾃﾞｱﾙ
230 A$=USR0("C6N83XNCCNZ")        :'OPEN
240 L$=""
250 IF USR7(0) THEN PRINT "GET END":END 'EOF
260 A=USR6(0):IF A=0 THEN 300             'LOC > 0?
270 A$=USR3(CHR$(1,0))
280 A$=RIGHT$(A$,1):A=ASC(A$)             'GET AND DISPLAY
290 IF (A>=&H20) OR (A=13) THEN PRINT A$;:GOTO 260
300 C$=INKEY$(0)                          'READ KEY
310 IF (C$=CHR$(0)) OR (L$=C$) THEN 340 'YUUKOU ?
320 IF C$=CHR$(13) THEN A$=USR5(""):GOTO 340
330 A$=USR4(C$)                           'TX IT
340 L$=C$:GOTO 250
```

その筋質問箱

へわたしゃ～真室川の解答者ァ～ヨイヨイである。

Q 私の使用機種はX1CとX1turboです。普段，turboばかり使っているのでCが泣いているではありませんか。そこで私はジョイスティックポートを使って，Cとturboをつないで，CをturboのRAMディスクとして立派に活躍させてあげました。しかし，わがままなCはもっとなにかしたいと言いました。私は悩んで，悩んだ末，グラディウスの100連発でもやってもらおうと思ったけど，うまく自機が動いてくれません。どうしたらいいのでしょうか。どうぞよいご指導を。

三重県　前川英紀

A ううむ，じつになつかしいことよ。というのは，その昔X1のハイパーオリンピックが全盛のころに，同じことをやっていたりしたからなのであった。

よーするにやっていることは，X1Cを連射機能のついた「ジョイスティックアダプタ」にしてしまおうということである。実際の構成は，ジョイスティック→X1C→両端がジョイスティックのコネクタになっているケーブル→X1 turboということになっている。そして，ジョイスティックからの信号をturboに渡す途中で，トリガをバシバシON/OFFしようということである。

さて，前川氏はマシン語プログラムのリストを送ってきたのであるが，それを見ると致命的な欠陥があるよーである。それはなにかというと，「速すぎる」ということである。すなわち，ゲームでは結局のところ100連発などというのは不可能で，せいぜい16～20連発ぐらいが限度なのである（本当はもっと少なくて5連発ぐらいかもしれない)。つまり，へたに速くするとゲームプログラムの側で読み飛ばしてしまったりするので，ちゃんと同期を取ってON/OFFしてやらないとうまく連射してくれないのである。そこで，とりあえずはBASICでもいいやということで作ったのがリスト1である。

リスト1　連射プログラムの基本

```
100 OUT &H1C00,7:OUT &H1B00,&B10111111
110 'MASK=&B10111111:RMASK=MASK XOR &HFF
120  MASK=&B11011111:RMASK=MASK XOR &HFF
130 OUT &H1C00,14:A=INP(&H1B00)
140 OUT &H1C00,15:OUT &H1B00,A AND MASK
150 OUT &H1C00,14:A=INP(&H1B00)
160 OUT &H1C00,15:OUT &H1B00,A OR  RMASK
170 GOTO 130
```

リスト2　究極の連射プログラム

```
100 OUT &H1C00,7:OUT &H1B00,&B10111111
110 'MASK=&B10111111:RMASK=MASK XOR &HFF
120  MASK=&B11011111:RMASK=MASK XOR &HFF
130 C=6:D=3
140 FOR I=1 TO C
150   OUT &H1C00,14:A=INP(&H1B00)
160   OUT &H1C00,15:OUT &H1B00,A AND MASK
170 NEXT
180 FOR I=1 TO D
190   OUT &H1C00,14:A=INP(&H1B00)
200   OUT &H1C00,15:OUT &H1B00,A OR  RMASK
210 NEXT:PRINT".";
220 GOTO 140
```

簡単に説明しておくと，100行でJOY1を入力，JOY2を出力に設定している。110，120行はトリガ1かトリガ2かを決めるビットマスクの設定である。ここではトリガ1(STRIG(n)で読み出されるほう）を連射するようになっている。130，140行はトリガをONにしてJOY2から送り出す部分（ビットをリセットするとONなのだ)，150，160行はトリガをOFFにしてJOY2から送り出す部分である。

このプログラムを走らせると，無条件でトリガ1が連射される。ウエイトを入れないで走らせると，だいたい毎秒40発ぐらいであるから，速すぎるので145，165行にFOR～NEXTの空ループをいれて調節することになる。しかしそれでは問題があるのだ。というのは，単純に空ループを実行しているだけだと，その間はスティックの動きをturboに伝えられないことになるからである。

よって，その筋のプログラムとしてはリスト2でまあまあ完成ということになるだろう。カウンタCとDの値を好みに応じて調節すると，あんばいがよくなるであろう。これ以上は，おそらく機械語でプログラムしても無駄であろうと思われる。なお，X1Cとturboをつなぐケーブルはいわゆるクロス接続になっている必要がある。では，次の方どーぞ。

Q 1．画面スクロール方法を教えろ。2．あのゼビウスはたぶんPCGでバックをスクロールさせていると思うのだが，よーく見ても4ドット単位なのである。どうしてだ？　3．PCG用のデータを作っておき，それを読んで定義する。1ドットずらしたデータを読んで定義する。これを高速に繰り返せばドット単位スクロールができるのでは？　以上，しっかりしっかり答えるように！

大阪府　藤田憲一

A むむむ，なんてなまいきな質問であろうことか。しかし，特に答えてやるからありがたく思えなのである。

「あのゼビウス」というのはX1用のゼビウスのことであろう。そう，確かにあれはバック（地面）が4ドット単位でスクロールするのである。そこで私はさっそくゼビウスの動作中にIPLスイッチを押してBASICを立ち上げ，抜く手も見せずに“DEFCHR TOOL”を走らせたのである。すると思ったとーりに，そこには「4ドットずれた地形図用パターン」が並んでいたのである。

そう，X1用のゼビウスはあらかじめ4ドットずれたA，B2セットのPCGを用意しておいて，「AグループのPCG→BグループのPCG→1行（8ドット）ずらしたAグループのPCG」という具合に表示して，4ドットスクロールを実現しているのである。実際には320×200の2画面をぎっこんばったんと切り換え表示しているのであろう。なかなかのハイテクと見た私である。

さて，「1ドットずらしたデータをPCG定義すればスクロールになるか」の質問であるが，理論的には可能であるが，実際はだめであろう。問題はよーするにPCGの定義速度である。turboで機械語を使えばそれなりのものは作れるかもしれないが，それでもかなり無理がある。だからあきらめたほうがよいであろう（数の問題もある)。

最後になったが「スクロール方法を教えろ」に対しては「教えません」と答えておくのである。教えてもらうことよりも，自分で解決していくことのほうがはるかに価値があるのだ。だから甘えるんじゃない。

てなところであるが，6月号で取り上げた「X1で2DDのドライブを使う」というのに関して，奈良県の芝脇岳雄氏をはじめ数人の方から解答が来たのである。普通はその解答を紹介したりするものなのであるが，実際は（物理＋論理）フォーマットプログラムの改造が必要だったりするので私は面倒臭くなってやめてしまうのである。そう，自分でバンバンしなさいなのである。てなところでその筋質問箱の最終回は暮れていくのであった。それでは，さよならさよならさよならである。

新・連・載

X68000BASIC入門—第1回

めぐりあいX-BASIC

（基礎編）

Nakamori Akira
中森　章

X68000のBASIC入門講座の始まりです。言語としての新しさと発展性がX-BASICの魅力ですが，第１回の今月はその雰囲気を理解してもらうために制御構造についてざっと目を通してみましょう。まずは記述の美しさに酔いしれてください。

あの日僕の脳天はハンマーでなぐられたようなショックを受けたのでした。そう，初めてX-BASICにめぐりあったときのことです。僕は，いままでBASICについて知っていたはずの常識が通用しないことに絶句したのです。

いままでのBASICはOSと言語の２つの性格を持ったコウモリ的言語でした。しかし，このX-BASICはもろに言語だったのです。しかも，MON, PEEK, POKE, DEVI$, DEVO$という低レベル（ハードに近い）命令は存在せず，まさに高級言語の顔で僕に微笑みかけてきたのです（ハード寄りの操作がまったくできないわけではなく，機械語で外部関数を作成することで実現できますが)。

そして，その文法は単純明快です。従来のBASICでは命令の数だけあるステートメントの書き方が文法でしたが，X-BASICでは命令が関数呼び出しで実行されるため，文法といえるものは変数の型宣言と制御構造の記述しかありません（関数の呼び方なんかは文法ではない)。また，関数が主体となることは言語の拡張を容易にします。むむっ，こいつは，と思った僕の気持ちがわかるでしょうか。ご存じのように，Xとは数学で未知数を表す記号です。そして，それを名前に冠したBASICは未知の小宇宙なのです。

転ばぬ先の型宣言

X-BASICで扱うことのできるデータ型は４バイト整数(int)，１バイト整数(char)，実数(float)，文字列(str)の４種類です。従来のBASICとの違いは，整数が２種類になった（１バイト整数が増えた）ことと，実数が１種類になった（単精度実数がなくなった）ことでしょうか。これらのデータを変数に入れて使用する場合，変数は型宣言をしなくてはなりません。もし，型宣言がされない場合はint型だとみなされてしまいます。これは従来のBASICが，明示的に型宣言がされない変数をすべて実数型とみなしていたことと同様です。

従来のBASICとは異なり，X-BASICでは変数名の最初の１文字で型を宣言すること（DEFINT, DEFDBLなど）も，%や#によって変数の型を指定することもできません。各変数ごとに，型を指定しなければならないのです。これは結構煩わしいことのように思えますが，プログラムで使用される変数を自分できっちりと管理するという意味ではとても大事なことなのです。X-BASICの型宣言はデータ型の後ろに変数名を並べるという方法で行います。たとえば，

```
int dragonar
char macross
float gundam
str ideon
```

という宣言はdragonarを４バイト整数型，macrossを１バイト整数型，gundamを実数型，ideonを文字列型変数として使用することを意味します。また同じデータ型の変数は変数名をカンマ(,)で区切ることによって一度に宣言することができます。たとえば，

```
int yuri,kei
```

ではint型の２つの変数yuriとkeiを宣言します。

ところで，変数宣言時に変数の初期値を設定することもできます。従来のBASICでは，定義していない変数への値の代入は，そのままその変数の宣言を意味していました(X-BASICでもint型の変数宣言になる)。たとえば，

```
momo=1
```

はmomoという変数（実数型）を宣言し，その値を１にする代入文でした。これと似たようなものでしょうか。変数の初期値を型宣言時に与えるためには，変数名の直後に＝を書き，それに続いて値を書きます。

```
int kazuya=100,tatsuya,minami=15
```

という記述は，３つのint型変数kazuya，tatsuya, minamiを宣言し，kazuyaには初期値100を，minamiには初期値15を与えることを意味します。tatsuyaは宣言するだけで初期値は与えていません。

また，従来のBASICと同様，各データ型の配列を宣言することができます。そのためには，dimを用いて，

```
dim char madoka(100)
dim int hikaru(100,200)
```

のように宣言します。すなわち，dimに続けてデータ型，配列名を書き，カッコで囲んで添え字の上限を指定します。上の例では，madokaは添え字が0から100であるchar型の１次元配列，hikaruは添え字が(0,0)から(100，200)であるint型の2次元配列を宣言しています。もちろん同じデータ型を持つ配列は，通常の変数と同様にカンマで区切って宣言することができます。たとえば，

```
dim float lum(10),ataru(20)
```

というぐあいです。

そして，X-BASICでは配列にも初期値を与えることができます。これは従来のBASICには見られない機能でしょう。一般にBASICで配列に初期値を与えるためには，

```
dim nantara(10)
```

という配列宣言のあと，

```
for i=0 to 10
    read nantara(i)
next i
```

などとして，READ文をFORループで繰り返し，DATA文から初期値を読み込んでやる必要があります。しかし，X-BASICではそのような面倒臭いことはしなくてよいのです。配列の初期値の設定も変数の場合と同じく，配列名のあとに＝を書き，配列要素をカンマで区切りながら，大カッコ｛と｝で囲んでやればよいのです。また，｛と｝で囲んだ文は１行に書かなくてもよいという決まりがありますから，配列の初期化は次のように，美しく行うことができます。

```
dim int seiya(5)={
    0,2,4,6,8,10
```

```
 }
 dim int hyouga(2,3)={
      0,1,2,3,
      4,5,6,7,
      8,9,10,11
 }
  dim int shun(1,2,3)={
      0,1,2,3,
      4,5,6,7,
      8,9,10,11,
      12,13,14,15,
      16,17,18,19,
      20,21,22,23
  }
```

上から順に1次元配列，2次元配列，3次元配列の初期化の例です。1次元配列のときは要素を順番に書けばよいのですが，2次元，3次元となるとどういう順序で要素を並べるのかわからなくなる人がいるかもしれません。そこで，ちょっと解説しておきましょう。

一般的にいって2次元配列

a(x,y)

が宣言されているとき，aは(y+1)個の要素を持つ1次元配列が(x+1)個の要素を持つ1次元配列になったものとみなすことができます。すなわち，上の例でhyougaという2次元配列は，

0,1,2,3

という要素を持つ1次元配列と，

4,5,6,7

という要素を持つ1次元配列と，

8,9,10,11

という要素を持つ1次元配列を並べたものということができます。

3次元配列の場合も同様に，

a(x,y,z)

は添え字の範囲が(y,z)で宣言された2次元配列を(x+1)個の要素とする1次元配列とみなすことができます。上のshunという3次元配列は，

```
0,1,2,3,
4,5,6,7,
8,9,10,11
```

という添え字の範囲が(2,3)である2次元配列と，

```
12,13,14,15,
16,17,18,19,
20,21,22,23
```

という添え字の範囲が(2,3)である2次元配列を並べたものと考えることができるのです。

3次元は2次元配列の並びであり，2次元配列は1次元配列の並びですから，結局はすべて1次元配列の初期値の並べ方に帰着してしまうのです。こう考えていくと多次元配列に初期値を与える場合に，どのような順序で要素を並べていったらいいかということがわかると思います。4次元以上は感覚的にとらえることができないのでよくわからないと思ったらもう負けているのです。

ステートメントを見たら制御構造と思え

X-BASICではほとんどすべての機能は関数呼び出しによって行われます。しかし，プログラムの流れを制御するような構造は関数では実現しにくいものです（余談ですがLISPは制御構造まで関数にしてしまっているゲゲボな言語です）。そして，こういう制御構造がX-BASICでの数少ないステートメント（関数でない命令）となって残っているようです。したがって，制御構造を理解してしまえば，X-BASICを8割がた理解したのも同然です。

それではその制御構造について，ザァーっと眺めてみることにしましょう。

if then else

まずよく使われるのがif-then-elseの構文です。これは従来のBASICにもありましたし，使い方もそんなに変わりません。これまでと同様，

if　条件式　then　文$_1$　else　文$_2$

の形式で使用します。“条件式”の値が真(0でない）なら“文$_1$”を実行し，“条件式”の値が偽（0）ならば“文$_2$”を実行します。また，else以降が存在しなければ“条件式”が真のときのみ“文$_1$”を実行します。これを流れ図で示すと図1のようになります。

このように従来のifとなんの違いもないのですが，X-BASICでは“文$_1$”，“文$_2$”を{と}で囲んで記述できる点が重要です。配列の初期化のところでも述べましたが，{と}で囲んだ記述は複数行にまたがってもよいのです。これは，従来のように，thenやelse間をマルチステートメントでぎちぎちに埋めて文を記述していたころと比べれば大きな進歩です（X-BASICでも：で区切ることによりマルチステートメントの記述が可能ですが)。やっとBASICのプログラムにも美しさを追及できる時代になったようですね。以下にif-then-elseの記述例を示します。

例[1]　1行で記述

```
if i=1 then print "いち" else if i=
2 then print "にい" else print "たく
さん"
```

例[2]　thenやelseの後ろを複数行に分けた

```
if i=1 then {
    print "いち"
}else{
    if i=2 then {
        print "にい"
    }else{
        print "たくさん"
    }
}
```

例[3]　elseとifをつなげてelse ifで使用

```
if i=1 then {
    print "いち"
}else if i=2 then {
    print "にい"
}else{
    print "たくさん"
}
```

行番号がいらない

まさかとは思いますが，プログラムの入力方法を知らない人はいないでしょうね。X-BASICでは行番号はプログラムの実行に無関係ですが，プログラムの入力のために必要です。つまり，プログラムの入力に関しては，従来どおりの

行番号　文

という形式で行います。これは，BASIC上でプログラムを入力する場合です。ただ，save，load命令によるセーブ，ロードがアスキー形式で行われますから，エディタ，ワープロ，ビジュアルシェルのNote Bookなどで作ったファイルをプログラムとしてロードすることも可能です。また，save@，load@命令によって行番号なしでセーブ，ロードすることも可能です。この命令を使えば，普通のテキストファイルをBASICのエディタ機能を使って編集することができるでしょう。

さて，ここでおもしろいバグ（？）を紹介しましょう。以下のプログラムをsave@命令でセーブしたのち，load@命令でロードしてみてください。

```
dim a(3)={
  0, 1, 2, 3
}
print a(0)
```

どうでしたか。save@命令でセーブしたのにもかかわらず，『行番号のあるファイルです』というエラーメッセージが出ましたね。それでは，ということでload命令を使うと，今度は『行番号のないファイルです』というエラーメッセージが出て，結局ファイルをロードすることができません。これは，上のプログラムの2行目が，

0, 1, 2, 3

であるために，この行が行番号付きの行（行番号は0）とみなされたためです。エラーチェックを厳しくしたために，使いやすいはずのload@命令がかえって使いにくくなっているという例でした。

for next

for nextのループもBASICではよく使われる制御構造です。これは，繰り返し回数がわかっている場合に，ループを作る制御構造でした。しかし，X-BASICのfor nextループは従来のBASICに対して少々制限が付け加わったものになっています。このステートメントは，

```
for 変数名 = 初期値 to 最終値
```

と，

```
next
```

との間を繰り返すもので，繰り返しは“変数名”で指定される変数を“初期値”に設定したのち，nextに到達するごとに変数を1だけ増加させていき，変数の値が“最終値”より大きくなるまで行われます。これを流れ図で示すと図2のようになります。

ただし，X-BASICでの以下のような制限があります。まず，“変数名”はint型でなければなりません。また，1ループごとの変数の増分を1以外に変える機能はありません。特に後者の制限はプログラムを作る上では困りものですね（そういう機能がなくてもプログラムは作れるが，美しくないものになる)。例としては以下のとおりです。

例1　単なる繰り返し

```
for i=1 to 4
    print "ここに来た"
next
```

この例では変数iが1から4になるまで繰り返しが行われる。すなわち，4回“ここに来た”がプリントされる。

例2　ループ変数の値を利用する

```
for i=1 to 4
    if i=1 then {
        print "いち"
    } else {
        print "その他"
    }
next
```

ループ変数の値はループ内で参照することができる。この例ではiの値が1のときだけ“いち”をプリントし，それ以外では“その他”をプリントする。

while endwhile

whileによるループはfor next のループとは異なり，繰り返し回数が不定の場合に用いるものです。このループも従来の BASICを知っている人には馴染みの深いものでしょう。これは，

```
while 条件式
```

と，

```
endwhile
```

の間を繰り返すもので，繰り返しは“条件式”が真である間実行されます(ループの終わりがwendでないことに注意)。このとき“条件式”の値は繰り返しの内容を実行する前に調べられます。これを流れ図で示すと図3のようになります。また，この while は無限ループを作るときにもよく用いられるようです。そのためには，常に真（0でない）ような式を“条件式”に与えてやればよいのです。たとえば，1（は0でない)を使って，

```
while 1
```

とすればよいのです。こうして作った無限ループからの脱出にはあとで述べる break を用います。これは，制御をプログラムの最初のほうに移すgoto(後述)の代わりに使用できますね。使用例は以下のとおりです。

例1　普通のwhile

```
i=1
while i>0
    input "数字";i
    print "まだまだ"
endwhile
```

この例では変数iに入力される値が正であり続ける限り“まだまだ”をプリントし続ける。

例2　無限ループ

```
while 1
    input "数字";i
    print "まだまだ"
    if i<=0 then break
endwhile
```

これは例1と同じ作用をする。

repeat until

このrepeat untilによるループもwhileと同じく，繰り返し回数が不定の繰り返しに用いられます。これは，

```
repeat
```

と，

```
until 条件式
```

の間を“条件式”が真になるまで繰り返すものです。“条件式”の値は繰り返すべき文を実行したあとに調べられます。これの流れ図は図4のようになります。while との違いは文を1回実行するかどうかということになります。whileが「～の間」という意味，untilが「～まで」という意味であることを知っていれば違いは明らかでしょう。

例1

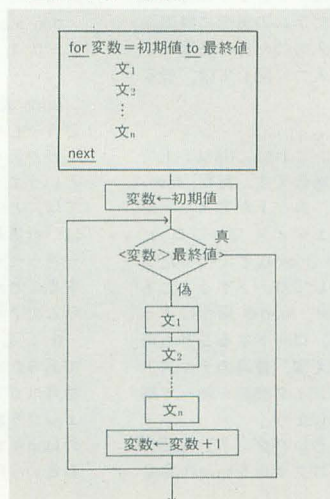
```
repeat
    input "数字";i
    print "まだまだ"
until i<=0
```

この例もwhileでの例と同じ。

switch case default endswitch

これは従来のBASICではあまり見ることのできない制御構造で，次のような形式で

▼図1 if then else

▼図2 for next

▼図3 while endwhile

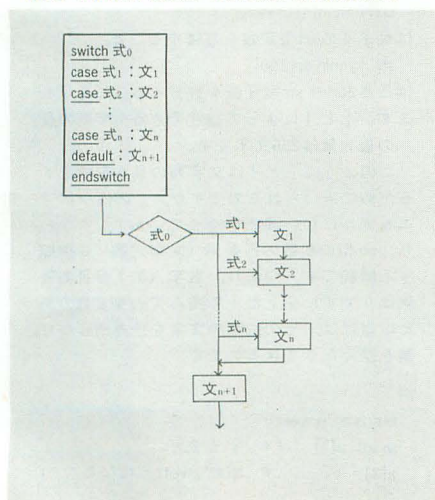
▼図4 repeat until

使用します。

```
switch 式0
case 式1  : 文1
case 式2  : 文2
・・・・・
case 式n  : 文n
default   : 文n+1
endswitch
```

ここで“式1”から“式n”までは整数か文字定数（1文字をアポストロフィで囲んだもので，その文字コードを値とする1バイト整数）でなければなりません。プログラムの流れは，“式0”の値を求め，もし，それと一致するものが“式1”から“式n”までの中にあれば，そのcase以降を実行するというものです。もし，“式0”と一致するものがなければ，default以降が実行されます。この動作を流れ図にすると図5のようになります。

なお，“式0”が“式1”から“式n”までと一致した場合でも，プログラムが上から順番に実行されてくる以上，default以降は必ず実行されることには注意が必要です。しかし，多くの場合はdefaultにたどりつくまでにbreak（後述）によってswitchとendswitchの間を抜け出してしまうため，default以降だけでなく，他のcase以降に書かれている文すら実行することはありません。また，default以降は省略してもかまいません。そのときは，“式1”から“式n”までに“式0”と一致するものがないと，なにもしないでswitchとendswitchの間を抜け出してしまいます。

PASCALのCASE文では，並べられた式と一致するものがなければエラーになってしまいますが，default（既定値）という選択肢を設けた点，あるいは選択肢にないものは無視されるという点はプログラムの自由度を高くしてくれるでしょう。さらに付け加えるならば，UNIX上のコンパイラ作成ツールyacc（先月の有田さんの記事にありましたね）によって出力されるプログラム（C言語）は，これと同様のswitch-case-defaultを用いて状態を巧みに変化させていくものなのです。

▼図5 switch case default endswitch

いままでのBASICで見ることのできなかったこのswitch-case-default-endswitchは，なにも奇をてらったものではなく，それが便利な制御構造であるから採用されたのだということはよく理解しておく必要があります。ところで，caseの式の次の：はマルチステートメントを示すものですから，caseの式の後ろになにも書かないときは：を付けてはなりません。以下にswitchを使用した例を示します。

例1 breakしない

```
switch i
case 1 : print "いち"
case 2 : print "にい"
default : print "たくさん"
endswitch
```

この例ではiが1のときは，“いち”，“にい”，“たくさん”がプリントされ，iが2のときは，“にい”，“たくさん”がプリントされる。それ以外では“たくさん”がプリントされる。

例2 defaultがない

```
switch i
case 1 : print "いち" : break
case 2 : print "にい" : break
endswitch
```

この例ではiが1のときは“いち”がプリントされ，iが2のときは“にい”がプリントされる。それ以外ではなにもプリントされない。

例3 文字定数を使用

```
switch c
case '1' : print "いち" : break
case '2' : print "にい" : break
default : print "たくさん"
endswitch
```

この例ではcの値が'1'（31H）のとき“いち”がプリントされ，cの値が'2'（32H）のとき“にい”がプリントされる。それ以外のときは“たくさん”がプリントされる。

例4 case式の後ろに：がない

```
switch i
case '1'
        print "いち"
        break
case '2'
        print "にい"
        break
default
        print "たくさん"
endswitch
```

この例は例3と同一の動きをする。

break

従来のBASICではfor-nextループやwhileループから抜け出すのにgoto文が許されていました。X-BASICではループからただちに抜け出すための制御構造としてbreakが用意されています。breakが実行されたときは，いちばん内側にあるfor-nextループ，while-endwhileループ，repeat-untilループ，switch-endswitchのうちのひとつを抜け出すのに使用します。以下に例を示します。

例1 単純なbreak

```
i=0
while 1
    if i=10 then {
        print "とお"
        break
    }
    i=i+1
endwhile
```

この例ではiの値が10になったときに，“とお”をプリントしてwhileによる無限ループを抜け出す。

例2 ループの入れ子

```
while 1
    for i=0 to 20
        if i=10 then {
            print "とお"
            break
        }
    next
endwhile
```

この例ではiが10のとき，“とお”とプリントし，breakによってfor-nextループを抜け出すが，その外側のwhileによる無限ループまでは抜け出さないので，再び同じ動作を繰り返す。

continue

continueはループをスキップするための制御構造です。先のbreakはそれ以降の命令を無視してループを終わらせるためのものでしたが，continueはそれ以降の命令を無視したいが，ループは抜けたくないという場合に使用します。for-next，while，repeat-untilのループの中でこのcontinueに出会うとcontinue以下は実行されず，ループの先頭に戻って次のループが開始されます（for-nextでは変数が1だけ増加し，whileまたはrepeat-untilでは条件式が調べられる）。以下に例を示します。

例1

```
for i=0 to 10
    print "i=";i
    if i=5 then continue
    print "ここに来た"
next
```

この例ではiが5のとき，continue以下にあるprint命令がスキップされるため，"ここに来た"がプリントされない。

例2

```
for i=0 to 5
  switch i
  case 1
      print "いち"
      continue
  default
      print "その他"
  endswitch
next
```

この例ではiが1のときには"いち"をプリントし，それ以外では"その他"をプリントするはずなのだが。不幸にもswitchの中でcontinueを使うとエラーになってしまう。これは，はっきりいってX-BASICのバグ，あるいは限りなくバグに近い制限といえる。

goto

善良なプログラマはgotoなんか使ってはなりません。ここでの解説も省きます。

gosub

X-BASICでは関数を定義できますから，サブルーチンは不要です。また，このgosubには制限が多すぎます。gosubもgotoと同じく禁じ手の制御構造です。

関数主体のプログラム

X-BASICは，

名前（変数，変数，…）

といった形式の関数呼び出しによって，ほとんどすべての仕事が行われます。したがって，こういった関数を並べること，あるいは，新たな関数を作って利用することがプログラミングという作業になります。ユーザーが望む関数がすべて用意されていることはまずありませんから，関数を定義する方法が重要になってくるのです。グラフィック関数やFM音源関数などと同様のマシン語の外部関数という形で関数を定義することも難しくありません（今月号34ページの「X-BASICの外部関数を作る」に詳しく載っています）が，ここではプログラム中で定義される関数について解説してみたいと思います。

関数の定義は，

func 戻り値の型　関数名(引数$_1$，引数$_2$，…)

という形式の文によって開始され，

endfunc

という文によって終わります。従来のBASICでは，関数は，

DEF FNナントカ

という形式で定義していました。これは関数の定義を1文で終わらさなければならないというひどいものでしたが，X-BASICではそのような欠点は解決されています。

ここで，"戻り値の型"とは，関数によって返される値（関数の値）のデータ型です。また，"戻り値の型"は，それがint型の場合は省略することができます。

次に，"引数"の指定方法を示します。これは，

引数名；引数の型

という形式です。たとえば，

func int zz(x;int,y;str)

という文はint型の引数xとstr型の引数yを持ち，int型を戻り値とする関数zzを定義することを示します。例によって引数がint型の場合は；以下を省略することができます。たとえば，上の例は，

func int zz(x,y;str)

と同じです。関数から値を返すためにはreturnを用います。returnは，

return(戻り値)

という形式で使用します。"戻り値"を(と)で囲むのを忘れないようにしましょう。それでは，以下に完全な関数の例を示します。

例1

```
func float average(x;float,y;float)
    return((x+y)/2)
endfunc
```

この例では，float型の引数xとfloat型の引数yから，2つの平均を計算してその値（float型）を返す関数averageを定義している。

例2

```
func str japanese(num)
  switch i
  case 0: return ("れい")
  case 1: return ("いち")
  case 2: return ("にい")
  default: return ("たくさん")
  endswitch
endfunc
```

この例ではint型（宣言は省略してある）の引数numに対応する日本語を示す文字列（str型）を値とする関数japaneseを定義している。

さて，X-BASICの関数は従来のBASICでいうサブルーチンとしても利用できます。これは値を返さない関数（戻り値のない関数）として位置づけられます。このとき"戻り値の型"というものは存在しませんから，それを省略して関数を定義するようにします。このような関数ではendfuncに到達すると自動的に関数が呼ばれた次の文に制御が移ります。また，endfuncに到達する以前に，強制的に関数を終了させるためには，

return()

という表現を用います。これは，なにもない値を返すということです（gosubによるサブルーチンからの戻りに使われるreturnとは別物です）。

一般的にいって，戻り値のない関数は，グラフィック関係や，サウンド関係，あるいはプリントを含む関数など，その値よりも関数の作用（絵を描く，音を鳴らす，プリントする）のほうが大事な場合に作られることが多いようです。以下に値を返さない関数の例を示します。

例3

```
func introduce(nam;str)
  print "私は";nam;"です"
endfunc
```

この例では文字列（str型）で与えられる引数namの前後に"私は"と"です"を付けてプリントする関数introduceを定義している。

＊　＊　＊

以上，駆け足でX-BASICの基礎となる事項（型宣言，制御構造，関数定義）を解説してきましたが，どうだったでしょうか。関数の定義についてはもう少し解説しなければならないのですが，それは次回にやりたいと思います。では，来月まで。

マニュアルにない機能

str型の変数を宣言すると，その文字列の長さは最大32文字になります。しかし，この長さは[と]とで明示的に指定できるようになっています。たとえば，

str nanno[4]

は長さ4のsrt型変数を宣言すること，

str kyonkyon[200]

は長さ200のstr型変数を宣言することを示します。[と]によって指定できる文字列の長さの最大値は256文字です。

このように[と]は文字列の長さを指定するために用いられるのですが，このほかにマニュアルにない機能を持っています。すなわち，str型の変数の各要素（文字定数）を参照する機能です。つまり，数字（第1番目の要素は0です）を[と]で囲んでstr型変数のあとに書けば，その位置の文字を読み出したり，書き換えたりできるのです。

例

```
str a="qwerty"
print a[3]  :/* 'r'を表示
a[3]='t'    :/* aは"qwetty"になる
```

BASICリレー連載 ―――――― プログラミング実況中継・4回表

愛はBASICを育てる

Shimizu Kazuto
清水 和人

四番サード清水和人。ファインプレーもすればトンネルもする，三振もすればホームランも打つ，過激なプレーが信条だ。しかしそればかりではない。――私にBASICを与えてくださった神様に感謝します――“愛”こそが彼のパワーの源なのだ。

数年前から，自分のBASICをコインロッカーに置き去りにするパソコニストが現れ始めた。常人にはちょっと考えられない。これも都会という生き残りゲームの一場面なのだろうか。自分の指を痛めて育てたBASICを捨てる親，私たちはそんな人にならないよう，信念と愛情を持ってBASICを育てよう。

科学技術分野で30年間王者として君臨している言語FORTRANはその文法が統一されており頼もしい限りだが，対してパソコンの世界のBASICはある程度の統一性を保ちつつも機種によって個性的である。正確無比かつ冷静なゴルゴ13を FORTRAN とすれば，BASICは人間味溢れるインストラクタ，ジェド豪士のような存在なのだ。どちらも魅力的であることには変わりないが，かといってゴルゴ13に人間味がないとはいえない（なにせ連載では現在最長不倒である）。BASICのプログラマは非常に柔軟である。好きでやってるからかもしれない。FORTRANのプログラマは……，じつはこれも柔軟なのである（少なくとも力量のあるプログラマは，だが）。

うーん，すでになにをいってるかわからなくなってきたぞ。まあ要するにいろんなBASICがあって，パソコニストはそれぞれのBASICを愛してるってことなんだな。ハイそうです。私もわがターボのBASICを愛しています。では私の一風変わった“BASIC育て”を紹介しよう。

切り札はこのコマンド

「まあ，まったりとしているのに全然しつこくないわあ。この秘密はなんなの？」お馴じみ栗田ゆう子嬢の決まり文句である（このマンガも単行本が10巻になり，ストーリーが水戸黄門のようになってきた）。

さて，誰もが自分の得意なプログラミングスタイルを持っているものだが，私の場合隠し味のコマンドがひとつだけある。これを使うと私も，わがBASICも生き生きとして躍動感を増す。まさに究極の調味料であるが，それはFOR～NEXTでも，GOSUBでもない（私は自分のためだけにプログラムを組むからON～ERRORなどもめったに使わない）。それはBASIC界の異端児，8ビット界の底抜け脱線ゲームの金原二郎と呼ばれる伝説のコマンド，

KEY0

なのぢゃよ。

「えーっうっそー，変なやつー」と思った人も多いだろう。KEY0はそれくらい異端児なんだが，その底力は本物だ。

この命令はHuBASICとBASIC-M25/S25にある。MZ-700のマニュアルには載っていないがちゃーんと使えるのだ。使い方は

KEY0，文字列

としておくと，キー入力待ちになったときに自動的に文字列が入力される。たとえば

A$=“FILES”+CHR$(13)⏎

KEY0，A$⏎

とすると，直接キーボードで

FILES⏎

と入力する動作が自動的に行われる。まあBASICを起動していろいろやってみてくださいな。「こいつあ使える」ってことがわかったでやんしょう？　またこんなこともできるんでっせ。

```
10 KEY0，“RUN”+CHR$(13)
20 END
```

として走らせてみるってえと，走るわ走るわ，ほっといたら一生走りまっせ。

これが原型だ

さて，このKEY0を使ってどんなことをやるのかだが，まあすぐ思いつくのがリスト1のようなプログラムである。たったの3行で30行はいらないし，マルチステートメントにすりゃ1行だ。なんのプログラムかって？　やってみりゃわかるのだ。RUNするとカーソルが点滅する。ここで PRINT“Oh！MZ”でもLISTでもFILESでも入力してみてくれなはれ。そのとおりのコマンドが画面表示のあと実行され，再びこのプログラムが走るのだ。んーすごい。これがBASICによるBASICインタプリタだあ！なんてことあるわけないが，とにかく入力されたものを実行するプログラムができたわけだ。「これは使える！」，いや「使える可能性がある！」といったほうがよいかもしれない。リスト1だけじゃなにもできないが，ほらアイデアがわいてくるだよ。

わいてこない場合の私は，このプログラムでしばし遊んでいるわけだ。そのうちこのプログラムの弱点を見つけた。

「リスト1は水に弱い！」

ではなくて

「リスト1はRUNに弱い」

のであった（ちなみにやってみると止まらなくなるよ）。なぜこうなるかというと，20行の後ろの“RUN”が，次のRUNの10行でLINE INPUTのA$に入るからなのだ。おー，美しい。なんとなくリカーシブ？

大リーグボール2号は風にも弱いのだが，

リスト1　原型である（全機種）

```
10 LINE INPUT A$
20 KEY0,A$+CHR$(13)+"RUN"+CHR$(13)
30 END
```

リスト1はじつは弱点だらけである。やれエラーに弱い，LOADに弱い，NEWに弱い。しかし，弱点を消そうとして工夫を凝らすのはどうだろう。この美しいまとまりを無用な行で難解にするよりは，このまま先に進んだほうがより本質をついているのではないか。とかいって本当はめんどくさいくせにいいわけをしつつ，先に進んでしまおう。

BASICの教育的指導

しばらく遊んでいるうちに飽きてきたので，コマンドでも追加しようということになる。でもってリスト2である。

なにやら急に行数が増えたが，内容は単純である。A$をやめて配列REC$の中に入力したコマンドを蓄えていく。そしてRECALLと入力すると30021行で分岐して30200行へ飛び，今までの入力を表示するのである。これによって今までの作業が手に取るようにわかる。うーん，どことなくヒストリ！

30140行を見てほしい。RUNがGOTO 30020になっている。RUNしてしまうと配列がクリアされてしまうのでこうなった。私の実力ではこのアイデアを思いつくのに1秒も遅れを生じてしまった。本来なら思いつく前に指が動かなくてはだめである。まだ修行が浅い。

ちなみに私は段下げとコメントが嫌いである。ましてや10数行のプログラム程度，コメントなしで理解できぬようでは「たわけっ，この未熟ものめがっ！」といわれてもしようがない。

もひとつちなみに，RECALLというのはどこぞのOSについてる機能である。それを使ってるうちに，X1に向かってでもクセでRECALLとしたくなるようになってしまった。それではってんで，なんとかかんとか考えたのがリスト2なわけだ。しからば，これはBASICの上にOSを作っていることになる。人間はわがままだから，「BASICの上でUNIXを走らせろ！」とかいう奴もいるかもしれない(いないいない)。私はそのさわりの部分をやっているのだ。「夢は大きく持とう」，これはプログラミングの第一条である。

リスト1はLOADに弱かったが，リスト2はLOADもできる。ただし，LOADされるプログラムにもリスト2がついていなくてはならない。つまり，この部分をつけておくとLOADされたほうの30020行へ飛んでくれるという寸法である。そのために30000行からのプログラムにしてあるのだ。試しにこのプログラム自身を

SAVE“LIST-2”↵

LOAD“LIST-2”↵

とすればその動作が理解できるだろう。ただしその際，配列・変数はクリアされてしまう。うーん勉強になるなあ。こうしてBASICの深さを理解していく，これこそがBASICの教育的指導である。

遊ぶことが王道だ

BASICはいろいろ使っているうちにわかってくる。本を読んでも覚えられないし，ましてや奥義，裏技のたぐいにはとても到達できない。プログラムもそうだ。使っているうちに長所・短所がわかってきて，さらに良いものに改造できるようになるのだ。かめばかむほど味が出る。スルメみたいなものなのだ。

さて，リスト2で遊んでいるうちに，プログラムも入力できることがわかる。たとえば，

10　PRINT “X1turbo”↵

と打ち込むとまさにその行が入力できる。LISTコマンドで見てみると，確かに10行が加わっている。うーん自分で成長する自己成長プログラムなのだ(30000行以降を入力しちゃうとバグが出るので気をつけて！)。さらにRECALLをやってみると，ちゃんと打ち込んだ行が登録されている。してみるとこれはもうエディタ気分。

リスト2　どことなくヒストリ（MZ-700/1500は不可）

```
30000 DIM REC$(100)
30010 REC=1
30020 LINE INPUT REC$
30021 IF REC$="RECALL" GOTO 30200
30100 FOR IREC=REC TO 2 STEP -1
30110 REC$(IREC)=REC$(IREC-1)
30120 NEXT IREC
30130 REC$(1)=REC$:REC=REC+1
30140 KEY0,REC$+CHR$(13)+"GOTO 30020"+CHR$(13)
30150 END
30200 FOR IREC=REC-1 TO 1 STEP -1
30210 PRINT IREC;":";REC$(IREC)
30220 NEXT IREC
30230 GOTO 30020
```

作業の記録が残るところがたまらない。

コマンドをいくらでも追加できるように30021行と30100行の間が空いているのだ。ここにいっぱいコマンドを突っ込んで，GOSUBやGOTOで分岐処理をする。これがこのプログラムの基本コンセプトなのだ。これこそわが子を育てる親心，何カ月も何年もじっくり寝かせたこの1本を作るための受け皿である。1日1コマンド，3日で3コマンド，3コマンド増やして2コマンド減らす。人生はワンツーパンチである。

ところがである。バグがよく出るプログラムではあるが，どれもたいしたことないと思っていたら，本質にかかわる「根本バグ」がひとつあったのだ。ガーン，思わず「アタックNo.1のテーマ」がフェードインしてしまったりするではないか。

いろいろ遊んでいるうちにわかったのであるが，ちょっと長い行を打ち込んだときに途中までしか表示されず，プログラムが止まってしまうのであった。やったね，そうこなくっちゃバグこそ明日への糧，というわけでBASICマニュアルを見てみる。

な，なんとKEY0に許される文字列の長さは63文字までだってえ（MZ-2500は31文字，MZ-700/1500となると15文字)。これは大変，どうしたらよいのだろう。例によって飛雄馬は落ち込んでしまう。ちょうど大リーグボール1号が打たれてしまったときのような状況だ。こんな場合私は，歩いているときも，食べているときも，夢を見ているときでも，このバグのことで頭がいっぱいになってしまう。そして星飛雄馬と同様に，マンションの屋上でぼんやり子供のマリつきを見ていたりするのである。やっとひらめいたのは1カ月もたってから……，なんてことはなくて，5秒後にはいちおうのメドがたっていた。

アイデアこそプログラム源

KEY0では文字列の長さに限度がある，これはわかった。ではどうするのか。KEY0以外のもので代用するのか，KEY0を使って特殊な処理をするのか。いろんなアイデアが頭の中に浮かぶ。

1)　KEY0の中にKEY0を入れる

これは少し考えるとダメそうなことがわかる。事実，実験してみるとダメである。

2) 長い行を分割してつなげる

何回かKEY0をしてはリストをつなげていく方法だが難しそうだ。

3) 行を画面に書いておき，カーソルをそこに移動してKEY0でリターンさせる

うーんこれははたしてうまくいくのだろうか。こんなことで実際の入力と同じことが起きるのだろうか。

3)が有望なのでいろいろテストしてみたところ，なんとかうまくいくようである。私は天才じゃないか，とつぶやきながら改造してできたのがリスト3というわけだ。今度は本格的でっせえ！　まあちょっと走らせてみてくだせえよ。ね，いいでしょう。少し変な文字がちらつくがそれを除けばもうすっかり普通のBASICに見えてきそうである(依然としてエラーやRUNには弱いが)。しばしいろんなダイレクトコマンドを入力してそれを楽しんでほしい。

たとえば，LIST, FILES, SEARCH“文字列”,? TIME$, WIDTH40(80),SAVE“LIST-3”……，などとやってからRECALLすれば，なにをやったかのリストも出る。私はここで自分の作ったプログラムを眺め続ける。少しずつ一人前になっていくわが子の成長を見る親の気持ちとはこんなものかもしれない。

さて，30140行が新しいアイデアなのである。理解するためにはコントロールコード表が必要である。CHR$の中の引数は10進で記されたコントロールコードである。30は上向きカーソル移動，13はお馴じみ改行コード，26はコントロールZすなわちカーソル以下の画面クリアである。動作は次のようになる。

1)　30020行でLINE INPUTで入力された文字列までカーソルが戻り，その上で改行される。このとき入力された文字列が実行される。

2)　その後“GOTO 30020”としたところでCtrl-Zが実行され，それ以下の画面がクリアされる。これによって余計な文字が残っている場合のバグが取れる（試しに26を除いてやってみるとよくわかるよ)。

3)　そしてGOTO 30020が実行され，再びカーソルはGOTO 30020の表示まで戻ってCtrl-Zで表示を消す。まさに大怪盗ルパンの手口である。

こんな手順を踏んでいるから，ちょこっとずつ文字がちらついたりするのである。でもやっぱりいい感じだ。さすがは万能BASICの底力を見たといえよう。

こうなってくると，いったいこのプログラムが走っているのかどうかわからなくなることが多い。すぐに見分ける方法はリターンキーを押して“GOTO 30020”がちらついたら走行中である。

この改良によって大きなメリットができた。RECALLやLISTで出た行をそのまま入力に使えることだ。RECALL の場合は“数字:”をデリートキーで消す必要があるのでややメンドーだが，これは次の課題としよう。

リスト3　長い行でも大丈夫（全機種)

```
30000 DIM REC$(100)
30010 REC=1
30020 LINE INPUT REC$
30021 IF REC$="RECALL" GOTO 30200
30100 FOR IREC=REC TO 2 STEP -1
30110 REC$(IREC)=REC$(IREC-1)
30120 NEXT IREC
30130 REC$(1)=REC$:REC=REC+1
30140 KEY0,CHR$(30,30,13)+"G.30020"+CHR$(26,13,30,26)
30150 END
30200 FOR IREC=REC-1 TO 1 STEP -1
30210 PRINT IREC;":";REC$(IREC)
30220 NEXT IREC
30230 GOTO 30020
```

スランプぢゃ

リスト3を見ているうちにスランプになった。アイデアが浮かんでこない状態である。そんなときはプログラムを整理するとよい。

世の中のプログラマは2種類に分かれる。プログラムが好きな奴と嫌いな奴だ。前者にリスト3を見せれば一生懸命解読して「なるほど」といったあと，アドバイスのひとつもしてくれるだろう。ところが後者の江戸っ子に見せたりすると，

「てやんでい,コメントぐれえ入れろってんだ。ラベルを使いねえ。こんちくしょお,べらんめえ！」

とまくしたてられてしまう。

そういうわがまま君のために，スランプの無駄な時間を利用してプログラムを清書しよう（もっとももあまりやりすぎると短いプログラムだけに見苦しいが……)。書き直すことによって新しいアイデアがわいてくるかもしれない。

んでもってリスト4である。どうだ，見違えるほど長くなっちゃって。これは次の方針で整理した。

1)　メインはサブルーチンを呼ぶだけ
2)　サブルーチンは全部ラベルで
3)　サブルーチンの内容は段下げ
4)　各ルーチンとも100番の領域で
5)　RETURN，ENDは99のつく行で

まあ見やすくはなっただろう。ただし，間違ってもプログラムが好きなマニアに見せてはいけない。バカにされるのがオチ。プログラムが嫌いな人はこのぐらいラベルを使うとようやく見てくれる。なお，HuBASICでは“ラベル名”とだけなっているとGOSUB“ラベル名”を実行する。30010～30050行と30310行にはこれが使ってあるのでMZ-2500では適当に直してもらいたい。

整理しているとき思いついた改良もある。

1)　30320行。ただリターンキーを押してもRECALLの記録は残らなくなった。空白

リスト4　プログラムを整理する（以下MZ-2500はラベルを変更)

```
30000 'LIST-4
30010 "INITIALIZE"
30020 "INPUT"
30030 "COMMAND"
30040 "RECORD"
30050 "KEY0"
30099 END
30100 LABEL "INITIALIZE"
30110       DIM REC$(100)
30120       REC=1
30199       RETURN
30200 LABEL "INPUT"
30210       LINE INPUT REC$
30299       RETURN
30300 LABEL "COMMAND"
30310       IF REC$="RECALL" "RECALL"
30320       IF REC$="" RETURN 30020
30399       RETURN
30400 LABEL "RECORD"
30410       FOR IREC=REC TO 2 STEP -1
30420       REC$(IREC)=REC$(IREC-1)
30430       NEXT IREC
30440       REC$(1)=REC$:REC=REC+1
30499       RETURN
30500 LABEL "KEY0"
30510       KEY0,CHR$(30,30,13)+"G.30020"+CHR$(26,13,30,26)
30599       RETURN
31000 LABEL "RECALL"
31010       FOR IREC=REC-1 TO 1 STEP -1
31020       PRINT REC$(IREC)
31030       NEXT IREC
31099       RETURN 30020
```

を見てもしようがないからね

2) 30410行。同じことを2度以上続けて入力したら2度目以降はやはり記録に残さない

3) 31020行。RECALLで表示されていた“番号：”を取り去った。これにより，そこにカーソルを持っていけば，そのまま入力が可能となる

へへっ，また一歩進んだぜ，整理してみるもんだね。長くなったので1画面からはみ出たのはまずいけど，けっこう美しく書けてるだろ？　うっふん，なんとなくナルシストだわ♡

質素なメインがツキを呼ぶ

このプログラムの美しいところはなんといってもメインプログラムがラベルへのGOSUBだけからなっていることだ。最初にこんなようなメインを書いて始めることをトップダウン方式のプログラムと呼んだりする。じつはこのトップダウン，あとからあとから機能を追加するのにとても便利な手法である。そして，今度そのターゲットになりそうなのは30300行からの“COMMAND”ルーチン。ここにコマンドを追加すればどうにでも成長するのだ。このプログラムは，いわば受け皿が整った，すなわち義務教育を終え中学出たてのホヤホヤといったところである。

そうこうしているうちにまたまたひらめいてしまうのが私である。今度はBASICでプログラムを組んでいるときによくある失敗をなんとかしようというものだ。ある行を打ち込んだら前にその行番号だったところが消えてしまって「困った困った」なんてことがよくある。そんなときのために，行を上書きするときは前のリストを保存しておこうというのが新しい機能である。しかし，中間コードになってメモリ中にあるリストをどうやって取り出すのか。ここでもKEY0が活躍するのである。

リスト5が新しいバージョンである。“RECORD”の部分が拡張されていて，入力されたのがBASICの行であれば，以前に入っていた行を配列OLIST$に格納していくのだ。IOLISTが格納している行の数で，24行まで格納できるようになっている（それ以上は捨てていくのだ）。プログラムをよく見ればその行番号を取り出し，そこのLISTを取ってKEY0とLINE INPUTによってOLIST$にリストの文字列が入るのがわかるであろう。KEY0はこのように次に現れるINPUT文などにも効くのだ。まさにKEY0は人間の身代わりとなってキーボードからの入力をシミュレートするのである。マグマ大使の人間もどき，あるいはパーマンのコピーロボットのようなものである。

プログラムのほうは“INITIALIZE”の中でOLIST$の配列宣言とIOLISTのリセットが加わり，“COMMAND”の中にOLISTというのが加わっていることも注意したい。このOLISTコマンド，ラベル“OLIST”へ行き，古いリストが表示される。もちろんそこから次の入力もできるのだ。

ところでREC$もOLIST$も配列が24しかないが，これは画面に表示しきれないものを覚えていてもしようがないからだ。これでも長い行があるとはみ出してしまうが，そんな場合は直接配列をプリントするコマンドを入力すればOKである。納得したら使

リスト5　とりあえずの完成版

```
1 'Don't DELETE 30000-
30000 'LIST-5
30010 "INITIALIZE"
30020 "INPUT"
30030 "COMMAND"
30040 "RECORD"
30050 "KEY0"
30099 END
30100 LABEL "INITIALIZE"
30110         DIM REC$(24),OLIST$(24)
30120         REC=1:IOLIST=1
30130         KEY 3,"GOTO 30020"+CHR$(13)
30140         WIDTH 80
30199         RETURN
30200 LABEL "INPUT"
30210         COLOR 4,0
30220         LINE INPUT REC$
30230         COLOR 7,0
30299         RETURN
30300 LABEL "COMMAND"
30310         IF REC$="RECALL" "RECALL"
30320         IF REC$="" RETURN 30020
30330         IF REC$="OLIST" "OLIST"
30340         IF REC$="LIST" "LIST"
30350         IF REC$="NEW" "NEW"
30360         IF REC$="RUN" "RUN"
30399         RETURN
30400 LABEL "RECORD"
30410         IF REC$=REC$(1) GOTO 30499
30420         FOR IREC=REC TO 2 STEP -1
30430         REC$(IREC)=REC$(IREC-1)
30440         NEXT IREC
30450         REC$(1)=REC$:IF REC<24 THEN REC=REC+1
30455         REC1$=LEFT$(REC$,1)
30458         IF REC1$=" " THEN REC$=RIGHT$(REC$,LEN(REC$)-1):GOTO 30455
30460         IF REC1$>="0" AND REC1$<="9" ELSE 30499
30462         DUM=INSTR(REC$," "):IF DUM=0 THEN OLIST=VAL(REC$):GOTO 30465
30463         OLIST=VAL(LEFT$(REC$,INSTR(REC$," ")-1))
30465         PRINT "LIST";OLIST;CHR$(26)
30466         KEY0,CHR$(30,30,13)+"G.30467"+CHR$(13):END
30467         KEY0,CHR$(30,30,30,13):LINE INPUT OLIST$
30468         DUM$=STRING$(LEN(OLIST$) ¥ 80 +3,CHR$(30))+CHR$(26)
30469         KEY0,DUM$+"G.30470"+CHR$(13):END
30470         IF LEFT$(OLIST$,1)<"0" OR LEFT$(OLIST$,1)>"9" THEN 30485
30473         FOR IREC=IOLIST TO 2 STEP -1
30475         OLIST$(IREC)=OLIST$(IREC-1)
30478         NEXT IREC
30480         OLIST$(1)=OLIST$:PRINT CHR$(30);
30483         IF IOLIST<24 THEN IOLIST=IOLIST+1
30485         COLOR 4:PRINT CHR$(30,26)+REC$:COLOR 0
30499         RETURN
30500 LABEL "KEY0"
30510         KEY0,CHR$(30,30,13)+"G.30020"+CHR$(26,13,30,26)
30599         RETURN
31000 LABEL "RECALL"
31010         FOR IREC=REC-1 TO 1 STEP -1
31020         PRINT REC$(IREC)
31030         NEXT IREC
31099         RETURN 30020
31100 LABEL "OLIST"
31110         FOR IREC=IOLIST-1 TO 1 STEP -1
31120         PRINT OLIST$(IREC)
31130         NEXT IREC
31140         "RECORD"
31199         RETURN 30020
31200 LABEL "LIST"
31210         LIST 2-29999
31220         "RECORD"
31299         RETURN 30020
31300 LABEL "NEW"
31310         KEY0,"G.30000"+CHR$(13,30,26)
31320         DELETE 2-29999
31330         "RECORD"
31399         RETURN 30020
31400 LABEL "RUN"
31410         "RECORD"
31420         GOTO 1
31499         RETURN 30020
```

ってみよう。その素晴しさがひしひしと伝わってくる。これで入力ミスしてもすぐOLIST。BASICの標準機能にしたいくらいである。

注目すべきは，入力されたのがプログラムの行かどうかの判断部である。誰でも思いつくのだが，このへんにBASICの凄みがある。しかし，だんだんプログラムが大きくなってきた。フリーエリアが心配である。ツールばかりでかくて，作れるプログラムが小さくては勝手が悪いが，8ビットマシンの泣きどころだからしようがない。

RUNしても大丈夫

ここまでやってくると弱点が気になってしようがない人もいるだろう。リスト5ではOLISTのほかにいろいろ細かい変更がなされている。

まずLISTコマンドだが，30000行から以降は表示しないようにしよう。こうすればいちいちこのツールの顔をおがまなくてもよくなる。LIST2-29999にしてしまったのがラベル“LIST”のルーチンである。なお，第1行には注意書きのコメントが入っているが，これはあとのRUNコマンドの変更にからんでいる。

NEW コマンドはツール自体を消してしまうとやっかいなのでDELETE2-29999にしてしまったのが“NEW”である。NEWもLISTも最初に空白が入ったり，少しでも文字列が違えば本来の機能に戻る。これをきちんと処理するのもよいが，まあフリーエリアとの関連もあるし，ここはそう神経質にならなくてもよかろう。

さて問題のRUNであるが，これを無条件にGOTO1としてしまうのが“RUN”ルーチンである（この日のためのコメント行なわけだ)。本当にRUNしてしまうと最初のINPUT文で“GOTO 30020”が入力されてしまう，ああなんてこと，変数やなんかがすべて初期化されちゃうから今までの記録がパーになってしまうのだ。本体のプログラムのEND文のところをGOTO 30020にしておけば正常終了後もこのツールに戻ってくれる。エラーで止まったときはF3キーに“GOTO 30020⏎”が入っているので，それを押せばやはり復活できるのである。

これで重要なコマンドが少しずつ改良された。しかしどうしようもない制約を忘れてはいけない。それは本体のプログラムで，このツールで使用している変数を使っちゃだめよってことだ。A$とかIのような変数は使ってないのでめったにぶつかることはないが，気をつけてくれたまい。

遊び心も忘れずに

堅苦しいコマンドばかりつけてもしようがないという人のためにTVコマンドをつけてみた。X1/X1turboのBASICではダイレクトコマンドでCRT0などを入力したり，SHIFT＋“＝”を押したりしてもよいが，ちょっと便利にしたのがリスト6である。リスト5にMERGEしてもらいたい。使い方は次のとおりである。

1）TV⏎と入力する
2）X1のテレビがつく
3）数字の1,3,4,6,8と“/”，“－”が各チャンネルに対応している。“/”が10チャンネル，“－”が12チャンネルである（東京地方対応なのだ）
4）上記以外のキーでコンピュータ画面に戻る

X1/X1turbo ユーザーにとってはプログラミング中のナイター中継観戦は当たり前のことであるから，この機能もあって当たり前。キーボードがチャンネルスイッチに代わった感覚は，やってみると意外に便利である。このへんは個人の好みによるところも大きい。私はよくプログラムで5秒ぐらいで7チャンネルを切り換えて，すべての番組のあらましを把握する。現代の聖徳太子である。

チェンジコマンドで締め！

最後にリスト7だが，これはよくエディタにあるチェンジコマンド（文字列置き換え）のエミュレーションである。アイデアにつまってディスク上のファイルを使っているのでスピードは遅いが，使ってみるとたいへん便利である。変数名を全部一度に変えたりするのに利用できる。

手法は簡単で，アスキーファイルとしてプログラムをセーブしたあと，シーケンシャルに読み出して，INSTRを使って巧みに変換している。1行に複数個の変更箇所があって，しかも文字列が重なっているとミスが起きやすいものだ。そのへんの処理を味わってほしい。また，“RECORD”を呼ぶとき，変換する文字を参考につけているのもローバ心である。

というわけで，バグだらけのわがままひとりよがりプログラムが自己満足たけなわのうちにできあがった。これであとは思いつくたびに新機能をつけたり，改良を楽しめるのだ。これから幾多の難関が待ち受けていることだろう。しかしガンバレ大作少年，とくれば，負けるなジャイアントロボ，なのである。

リスト6　TVコマンドの追加（X1/X1turboのみ）

```
30370        IF REC$="TV" "TV"
31500 LABEL "TV"
31510        CRT 0
31520        DUM$=INKEY$:IF DUM$="" THEN 31520
31530        DUM=INSTR("13468/-",DUM$)
31540        IF DUM=0 THEN 31590
31550        IF DUM$="/" THEN DUM$="10"
31560        IF DUM$="-" THEN DUM$="12"
31570        CHANNEL VAL(DUM$):GOTO 31520
31590        CRT 1
31599        RETURN 30020
```

リスト7　チェンジコマンドの追加

```
30380        IF REC$="CHANGE" "CHANGE"
31600 LABEL "CHANGE"
31605        DUM=1
31610        INPUT "Input old text:",OLD$
31615        INPUT "Input new text:",CNEW$
31620        PRINT OLD$;"=>";CNEW$:INPUT "OK?(Y/N)",DUM$
31625        REC$=REC$+CHR$(13)+OLD$+"=>"+CNEW$:"RECORD"
31635        IF DUM$="N" OR DUM$="n" THEN 31610
31640        SAVE"DUM1",A
31645        OPEN"I",#1,"DUM1":OPEN"O",#2,"DUM2"
31650        LINE INPUT #1,DUM$
31655        DUM=INSTR(DUM,DUM$,OLD$)
31660        IF DUM=0 THEN 31675
31665        DUM$=LEFT$(DUM$,DUM-1)+CNEW$+RIGHT$(DUM$,LEN(DUM$)-DUM-LEN(OLD$)+1)
31670        DUM=DUM+LEN(CNEW$):GOTO 31655
31675        PRINT #2,DUM$:DUM=1
31680        IF EOF(1) ELSE 31650
31685        CLOSE #1:CLOSE#2:MERGE "DUM2"
31699        RETURN 30020
```

データと経験は真実を語る

Katsumoto Shin
勝本 信

再び地下室へ

アルミニウムを100万分の1センチ程度の微粒子にして，零下273度程度まで冷やしておく。微粒子のまわりにはコイルを巻きラジオ波を与える。外側から強力な電磁石を使って磁場を少しずつ変えていく。するとアルミニウムAl^{27}の原子核スピンがラジオ波と共鳴するところで，ラジオ波の吸収が起こり，コイル両端の電圧に変化が見られる。

図1に示したのは，実際にコイルの電圧変化を測定した結果である。コイルの両端につながれた位相敏感増幅器からの出力をGP-IBインタフェイスを通してコンピュータに取り込み分析を行った。黒い丸で示された点が実測データである。くねくねと曲がっており一見複雑な形だが，じつは2つの簡単な曲線の重ね合わせであったことがわかる。

我々は実験を行うことにより，自然や宇宙の仕組みを知ることができる。しかし，我々が実験によって得られるデータは自然の真の姿からはほど遠い。電子や原子核を直接見ることができない（最近はそうでもないが）のはもちろん，はるかかなたにある恒星や星雲に行くこともできない。いわば遠くから曇りガラスを通して見ることが許されるだけである。その上，どのような実験にも誤差がついてまわる。どんなに精確な実験を行ったとしても，あるいは何回も繰り返し測定を行って平均をとったとしても誤差を0にすることはできない。

今回は，測定データを分析することにより，曇りガラスの遠くにぼんやり見えていた宇宙の姿がはっきりと浮かび上がってくるという話である。

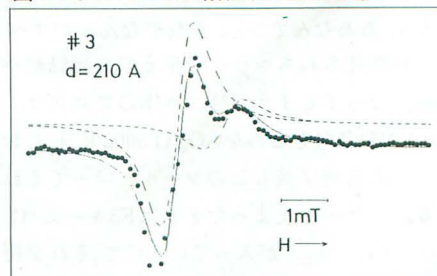
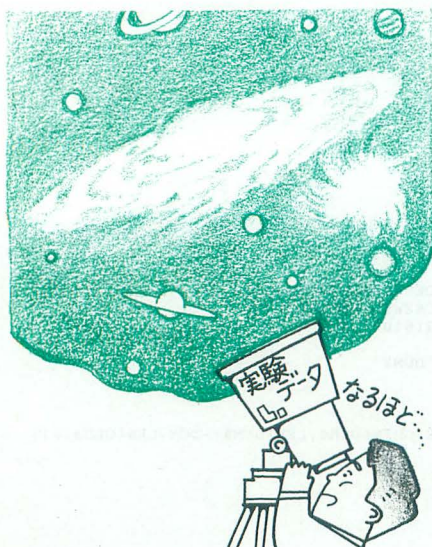
簡単な例として，バネの伸びとぶら下げた重りの関係を取り上げる。頭の中で実験を行ってみることにしよう。バネにいろいろな重さの重りをぶら下げて，その伸びを測定する。1グラムの重りを付けたときには何センチ伸びたかということを記録しておく。

次に結果をグラフにしてみる。横軸には重りの重さを，縦軸にはバネの伸びをとって点を打ってみる。すると多少のばらつきはあるものの，図2のようにデータは一直線の近辺に並ぶであろう。ここで各データ点を線で結んで折れ線グラフにしてしまったのでは自然の真の姿は見えてこない。

幸いにも我々はバネの伸びが重りの重さに比例するであろうということを経験として知っている。そこでできるだけ多くの測定点の近くを通るように直線を引いてみる。この直線の傾きが急なほど弱いバネ，傾きが緩やかなほど強いバネであるということになる。

この場合，バネの伸びが重りの重さに比例することを「知っていた」からこそ，ばらつきのあるデータに直線をあてはめることができたのだが，まったくなにも知らない状態で，図1のデータを見せられた場合はどのように対処すればよいのだろうか。そのときはもはや人間の直感で立ち向かうほかない。自分の持っている経験と直感とを駆使して，目の前にあるデータの意味を判断するのである。たとえば，このデータは一直線上に乗るに違いないと判断したならば，バネの種類や重りの重さの範囲を変えたりしてさらに実験を行ってみる。すると，バネの固さによって直線の傾きが異なることや，非常に重い重りの場合にはデータが同じ直線に乗らなくなる，などさまざまなことがわかってくるだろう。しかしどの測定結果も，重りがあまり重くない場合にはバネの伸びは重りの重さに比例するということを支持しているに違いない。

一方，判断が誤っていた場合はどうか。データは放物線上に乗るということが真実であるにもかかわらず，直線に乗るという法則を採用してしまった場合などである。測定誤差が大きく，データがばらついているときには，判断を誤ってしまうことも多いだろう。この場合にも測定の範囲や測定の対象を広げて幅広い実験を行えば，いつかはつじつまが合わなくなり，測定データに対する判断が誤っていたことがわかるはずである。

図1　アルミニウム微粒子の磁気共鳴

黒丸が実測データ。点線が2つのシグナルに分解したもの。実線はその和。

そうしたらまた別の新しい法則をあてはめてみればよい。実験を行ってその結果から法則を導き出し，あるいは理論を組み合わせて法則を予言し実験によってそれを検証する。科学はそうやって進歩してきたのである。

最小二乗法とのめぐりあい

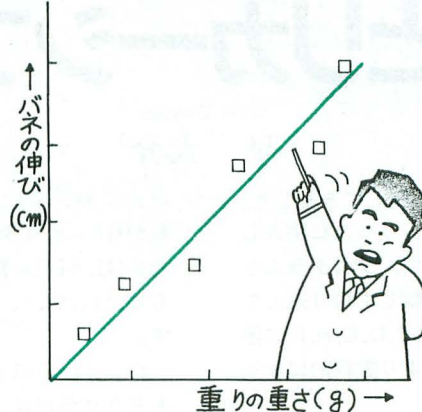
次に，実験データに法則をあてはめる方法について説明しよう。まず，例によってバネの伸びと重りの重さのデータに直線をあてはめることを考える。いちばん簡単なのは「目分量」で直線を引くことである。定規をあてて，できるだけ多くのデータ点の近くを通るように線を引けばよい。この方法は簡単であるうえに確実なのではあるが，正確さに欠ける。特に，直線の傾きを精密に求めたい場合などには不十分である。

その上，データにあてはめる法則が直線以外の場合には「目分量」はまったく使えない。放物線や対数曲線などの複雑な曲線を目分量で描けるわけがないからだ。そこで登場するのが最小二乗法であり，コンピュータが活躍することになる。

最小二乗法を用いると，測定データに対する直線や放物線のあてはめをきわめて精密に行うことができる。たとえば図2のように直線をあてはめる場合には，その直線を表す数式が求まる。直線に限られたわけではない。放物線や対数，指数，さらに図1のような複雑な曲線に対してもたちどころに対応する数式が求められる。

このすばらしい最小二乗法の仕組みを簡単に説明しておこう。データ点とあてはめた曲線とのずれを2乗したものをすべて足し合わせ，それが最小となるように曲線を定める，これが最小二乗法のすべてである。ようするにデータと曲線とのずれがもっとも少なくなるような曲線を求めるだけである。なぜ4乗や絶対値でなく，2乗したものの和を最小にするのかについてはきちんとした理由がある。測定で得たデータには必ず誤差があることはすでに述べたが，この誤差がxであるような確率密度は多くの場合e^{-x^2}という関数(正規分布, exp(−x^2)と書いたほうがわかりやすいかもしれない)に従う。この関数が最大になるところ，すなわちもっとも確からしい点が，xの2乗が最小になるところなのである。

実際に，ずれの2乗の和が最小になる点

図2　バネの伸びと重りの関係

を探す方法については，多種多様な方法が考案されている。たとえばもっとも安直な方法として，直線の傾きを少しずつ変えていって最小になる点を探してもいいのだがそれでは手間がかかりすぎる。

幸いにも偏微分という手法を用いると，直線や多項式の場合（正確に言えば線形の場合）には連立方程式を解くだけで曲線を決定できるということがわかっている。このプログラムはBASICでも簡単に書くことができるし，ポケットコンピュータの中にはPC-1450/1470Uなど，線形最小二乗法のプログラムを内蔵しているものさえある。身のまわりのデータを数値化して最小二乗法で分析してみるというのもおもしろいだろう。

本質を見極める

実は，図1に示したグラフの曲線は連立方程式を解くだけで決定できるという代物ではない。なにしろ曲線を表す関数の形が複雑である。言葉で言えば「2つのローレンツ曲線を微分したものの重ね合わせ」であり，式で書けば

$$\frac{a(x-b)}{((x-b)^2+c^2)^2}+\frac{d(x-e)}{((x-e)^2+f^2)^2}+g$$

となる。この曲線とデータとのずれを2乗して足し合わせたものを最小にするなどと聞くと一瞬気が遠くなるが，パーソナルコンピュータのおかげで，わずか数分で結果が得られる。プログラムがaからgまでの定数をいろいろと変えながら最小点を探してくれるのである。

探し方についてもいろいろ研究されており，もっともポピュラーなものは最急降下法と呼ばれるものである。曲線を表す数式の定数（この場合a～g）を少しだけ変化させたときに，ずれの2乗の和がどれだけ減少するか調べ，もっとも急激に減少するように定数を変えていく。これはちょうど谷底へ降りるときに，いちばん急な道を通って行くようなものだと考えればよいだろう。

このほかにももっと能率がよい方法がいくつか考案されており，たとえばフレッチャー・パウエル法や，マルカール法（どちらも考案者の名前）などが知られている。図1の曲線はマルカール法で決定したが，BASICコンパイラを用いて5分程度の計算量であった。余談になるが，この曲線はローレンツ曲線を微分したものを2つ重ね合わせたものであることはすでに述べた。このことはアルミニウム原子核スピンの共鳴点が2つあることを意味しており，微粒子の一部は超伝導状態に，残りは常伝導状態にあるためではないかと考えている。2つのローレンツ曲線の大きさの時間変化や磁場による変化を，最小二乗法を使って詳しく調べることにより初めて得られた予想である。

最小二乗法によるデータの解析について述べてきたが，必ずしも最小二乗法が万能であるとは限らない。放射線の計測など，誤差が正規分布しない場合には最小二乗法は使用できない。また，振動するデータの周期などを詳しく調べたいのであればフーリエ変換という手法を用いたほうが効率がよい。データ解析でもっとも基本的な手法のひとつが最小二乗法なのであると考えたらよいだろう。

なお，最小二乗法のプログラミングや数学的証明について興味があれば

データ解析――アナログとデジタル
（粟屋，学会出版センター）
最小二乗法よる実験データ解析
（中川・小柳，東大出版会）

などを参考にしてほしい。

海王星は1864年にルベリエとアダムスによって発見されたが，その際，最小二乗法を駆使して位置の予言を行い見事的中した。「一を聞いて百を知る」という諺があるが，データ解析を行うことにより，百どころか宇宙の本質さえ見極めることができるのである。

次回はファイルに関するプログラミングについて考える。

EXERCISE-20

マシン語体操1・2・3

万年暦とリカーシブコール

Izumi Daisuke
泉　大介

4月号から4回にわたってZ80の基礎知識を解説してきました。今月からはこれまでのHop to itを拡大して新シリーズに突入します。先月までに取り上げた20個の命令に加えて，プログラムを作るうえで便利な命令は，そのつど使い方を解説して取り込んでいく予定です。また，レジスタに関してもA,B,C,D,E,H,Lに限らずに，必要ならどんどん利用していきます。より実践的になったエクササイズでステップアップしましょう。

さて皆さん。自分の誕生日は何曜日だったのか，フランス革命のバスチーユ牢獄襲撃は何曜日だったのか，知りたいと思ったことはありませんか。任意の年のカレンダーを見ることができれば，こんな疑問もすぐに解決します。任意の年のカレンダーを表示する万年暦制作の第1段階として，先月はひと月分のカレンダーを表示するサブルーチンPRMONを作りました。今月はこのPRMONを使って万年暦を完成させましょう。

1日は何曜日？

万年暦を作るときに絶対に必要な知識として，調べたい月が何曜日から始まるのかを決定する方法があります。まず最初に1年が365日固定だとして，月の初めの曜日を決定する方法を考えてみましょう。

月の初めの曜日を決定するには西暦元年1月1日からの日数を調べ，それを7で割った余りを考えてやれば簡単です。西暦元年1月は月曜日から始まっていますから，7で割った余りが1なら月曜日，6なら土曜日だと判定できるわけです。たとえば，西暦m年のn月が何曜日から始まるのかを知りたければ，

(m−1)×365＋[n−1月までの日数]＋1

として，西暦元年からの日数を算出し，それを7で割った余りを取れば求まります。ただし，この方法では西暦179年の7月で日数がHLレジスタに入る最大値65535を越えてしまいますので，新たにHL×DE＝HLDEというような掛け算ルーチンを用意してやらなければならず面白くありません。そこで別の手を考えてみます。

(m−1)×365で西暦元年からm−1年までの日数を出してやってもどうせ最後は7で割って余りを調べるのですから，

(m−1)×(365 mod 7)＋[n−1月までの日数]＋1

として，これを7で割った余りを取っても結果は変わりません。都合のよいことに(365 mod 7)は1ですから，この式はさらに

(m−1)＋[n−1月までの日数]＋1

と簡単にすることができます。これならたとえ西暦5万年がきても大丈夫ですから十分実用になるといえるでしょう。

あとは閏年の処理を加えれば，任意の年の任意の月が何曜日から始まるのかを調べてやることができます。閏年なら1年の日数が1日増えますから，先の式にm年までの閏年の回数を加えてやり，それを7で割った余りを求めればいいのです。つまり，

(m−1)＋[m年までの閏年の回数]＋[n−1月までの日数]＋1

という式で求めた答えを7で割り，その余りを調べるわけです。

そこでm年までの閏年の回数を調べる方法です。閏年は

1)　西暦が4で割り切れるなら閏年
2)　ただし，100で割り切れるなら平年
3)　ところが，400で割り切れるなら閏年

という規則に従って巡ってきます。ですから，

Y1＝INT((m−1)/4)
Y2＝INT((m−1)/100)
Y3＝INT((m−1)/400)

とすれば「Y1−Y2＋Y3」がm年までの閏年の回数になります。たとえば，1987年までに何回の閏年があったのかをこれに従って計算すると，

Y1＝496，Y2＝19，Y3＝4

ですから(496−19＋4)回，すなわち481回の閏年があったということになるわけです。

以上の知識を使って，1987年の1月1日が何曜日なのかを調べてみましょう。まず(m−1)は1986ですね。そして1987年までの閏年の回数は481回。(n−1)月までの日数はいまnが1ですから0。そして最後に1を加えます。1986＋481＋1は2468。これを7で割ってその余りを求めると4になります。4は木曜日ですね。カレンダーを取り出して確認してみてください。確かに木曜日から始まっていますね。

1日の曜日を調べる方法がわかったところで次の題目に移ります。調べたい年と月をプログラム中に書き込んでアセンブルしてから実行するというのは簡単ではあるのですが，どうもマヌケです。BASICでプログラムを作るならもちろん実行時にキーボード

今月登場する命令たち(18語)

命令	説明
LD	値を入れる。「LD(9876H),A」で9876H番地にAが入る
CALL	サブルーチンを呼ぶ。「CALL NZ,#NL」はノンゼロなら#NLをコールする
RET	サブルーチンから帰る。「RET C」はキャリなら帰る
PUSH	スタックにレジスタの値を保存する。(ex.「PUSH HL」)
POP	スタックからレジスタに値を取り出す。(ex.「POP DE」)
XOR	A=A XOR m。mはレジスタまたは数値
OR	A=A OR m
CP	Aとmを比較する。結果はフラグに残る
ADD	A=A+m, HL=HL+rp。rpはレジスタペア (HL, DE, BC)
ADC	A=A+m+cy, HL=HL+rp+cy。cyはキャリなら1
SUB	A=A−m
SBC	A=A−m−cy, HL=HL−rp−cy。
INC	r=r+1。rはレジスタ (B, C, D, E, H, L, (HL), A)
DEC	r=r−1
JP	BASICのGOTOに相当。「JP 1FFDH」は1FFDHへジャンプする
JR	相対ジャンプを行う
DJNZ	「DEC B」「JR NZ,~」を1命令で行う。フラグの変化なし
EX	「EX (SP),HL」はスタックトップとHLレジスタの内容を交換する

から年と月を入力してもらうように作りますね。マシン語でもそうすることにしましょう。

#GETLルーチンを使う

S-OSにはキーボードから1行入力を行う #GETL というルーチンがあります。これはBASICのLINPUT（LINE INPUT）命令と同じことをするルーチンです。BASICの場合は文字変数に読み込みましたが，マシン語には文字変数などありませんから読み込むアドレスをレジスタで指定してやることになります。どのように取り込みが行われるのかは実際に試してみたほうがわかりやすいでしょう。次のプログラムをZIMPLで入力してください。

```
8000 11 00 90    LD     DE,$9000
8003 CD D3 1F    CALL   $1FD3
8006 C9          RET
```

まず，8000H番地で入力文字列を取り込むアドレスを指定します。次に，1FD3Hの #GETL ルーチンを呼び出して1行入力を行い終了です。

それでは実行してみてください。画面上でカーソルが点滅し入力を促していますね。ここで「012345」と入力してみましょう。入力はリターンキーで終了します。改行して次の行の先頭でカーソルが点滅していますね。ではここでマシン語モニタを使ってメモリの内容を確認してみます。画面上の1行を9000H番地から取り込むよう指定しましたから，メモリダンプコマンドで9000H番地の内容を表示させてみてください。

```
9000：30 31 32 33 34 35 00 ……
```

と表示されます。30というのは“0”のアスキーコード，31というのは“1”のアスキーコードというぐあいに，入力した文字に対するアスキーコードが順に入っています。そして7つめの00が文字列はここで終わりだヨというマークです。入力したのは確かに6文字でしたからこれで合っていますね。このようにキー入力用に割り当てたメモリのことをキー入力バッファと呼びます。

入力はリターンキーだけではなくブレイクキーで終了することもあります。このときにはキー入力バッファの先頭に1BHが書き込まれることになっています。もう一度このプログラムを実行し，今度はブレイクで終了してみてください。バッファの先頭である9000Hに1BHが入っているでしょう。

また，画面に「Input Year:」と表示して改行せずに西暦を入力させると，バッファには表示した「Input Year:」も取り込まれます。INPUTではなくLINPUTであることに注意してください。

このようにキーボードからの入力はすべてアスキーコードで入りますから，数値を入力するときには文字列を数値に直すルーチンを作ってやらなければなりません。では次に，文字列を数値に直す方法についてお話ししましょう。

数字を数値に変換する

まず最初に，ひと桁の数字を数値に変換する方法を考えてみましょう。アスキーコードでは“0”が30H，“9”が39Hというぐあいに順序よく数字が並んでいます。つまり，ひと桁の数字を数値に変換するには，数字のアスキーコードから30Hを引いてやればよいのです。たとえば，キー入力バッファをのぞいてみてそこに35Hが書き込んであれば，

```
LD   A,(DE)；キー入力バッファの先頭文字を取り出す
SUB  '0'    ；=SUB 30H
```

としてやることでAレジスタに対応する数値「5」を得ることができます。

2桁以上の数字を変換するときには次の方法を使います。まず最左桁を持ってきて，これを数値に変換します。これは上の方法でできますね。次にもうひと桁持ってきて，これが数字なら現在の結果を10倍した答えにいま持ってきた数字を数値に変換して足します。具体的にやってみましょう。「123」を変換してみます。

まず先頭の文字“1”を数値に直します。次の文字も数字ですからこれを10倍すると10。“2”を数値に直して足すと12です。さらに次の文字も数字ですから今と同じ手順で変換すると，12を10倍して120。これに“3”を数値に直して加えて123となります。文字列はこれで終了ですから答えは123ですね。

このとおりに入力文字列の数値変換をやっているのがリスト1です。このサブルーチンは数字列が入っている入力バッファの先頭アドレスをDEにセットして呼び出します。「#GETLルーチンを使う」で見たように1行入力の最後は00Hで終わることを思い出しておいてください。

まず，141行で答えを入れるレジスタHLを初期化します。そして142～147行でバッファから1文字取り出し，これが数字かどうかを判定します。取り出したデータが数字だった場合には148行にきます。ここでは現在の数HLを10倍し，それに取り出した文字を数値に変換して加えるという処理を行います。

「ADD HL,HL」は同じものを2度足しますから，これは2倍するのと同じことです。さらにもう一度「ADD HL,HL」を実行すると2倍したものを2倍するのですから，これは4倍するのと同じことです。同じ調子で8倍，16倍，32倍と，「ADD HL,HL」だけを使って簡単に作ってやることができますね。では10倍したいときにはどうすればいいでしょう。これは

$$10 \times HL = (8+2) \times HL = 8 \times HL + 2 \times HL$$

と変形して，2倍と8倍の和の形にしてやればいいのです。ではリストを見てください。148～150行でBCにHL×2を入れます。さらに「ADD HL,HL」を繰り返し，151，152行でHL=HL×8を求めてから先ほどBCに取っておいた「HL×2」を153行で加えれば，みごとHL=HL×10のできあがりです。

155行の時点でAにはバッファから取り出した数字が入っていますから，これから“0”のアスキーコードを引いて数字を数値に変

リスト1　数字を数値に変換する

```
80CC              132 ;  CONVERT CHAR TO INT
80CC              133 ;
80CC              134 ;  in   : DE = Top of Buffer
80CC              135 ;
80CC              136 ;  out  : HL = Number
80CC              137 ;
80CC              138 ;  brkn : AF,BC,DE
80CC              139 ;
80CC              140 CONV:
80CC 21 00 00     141         LD      HL,0
80CF 1A           142 CONV1:  LD      A,(DE)
80D0 13           143         INC     DE
80D1 FE 30        144         CP      '0'
80D3 38 12        145         JR      C,CONV2
80D5 FE 3A        146         CP      '9'+1
80D7 30 0E        147         JR      NC,CONV2
80D9 29           148         ADD     HL,HL          ; *2
80DA 4D           149         LD      C,L
80DB 44           150         LD      B,H            ; BC=HL*2
80DC 29           151         ADD     HL,HL          ; *4
80DD 29           152         ADD     HL,HL          ; *8
80DE 09           153         ADD     HL,BC          ; *10
80DF              154         ;
80DF D6 30        155         SUB     '0'
80E1 4F           156         LD      C,A
80E2 06 00        157         LD      B,0
80E4 09           158         ADD     HL,BC          ; HL=HL*10+A
80E5 18 E8        159         JR      CONV1
80E7              160 ;
80E7 B7           161 CONV2:  OR      A              ; Check End Code
80E8 C9           162         RET
80E9              163
```

換します。156～159行でこれをBCにコピーして10倍したHLに加え，次の文字を取り出して変換するためにループします。

次にバッファから取り出した文字が数字でなかったときの処理を考えます。もし取り出した文字が00Hだったならこれは入力文字列の終わりだということですから正常終了です。それ以外の文字だった場合にはエラーとします。この処理を行っているのが161行の「OR A」です。取り出した数字以外の文字が00Hならここでゼログラグが立ちますし，もしそれ以外の文字だったならノンゼロとなります。ですからメインルーチンでは

```
LD    DE, BUFFER ; バッファの先頭アドレスをセット
CALL  CONV       ; 変換ルーチンの呼び出し
JR    NZ, ERROR  ; ノンゼロならエラー
```

というぐあいに処理してやれば，入力してもらった数字が正しかったかどうかを判断できるわけです。

162行で正常終了か異常終了かのフラグを持ってリターンして，CONVルーチンは終了します。

メインルーチンの制作

ではいよいよメインルーチンの制作に取りかかりましょう。カレンダープログラムの処理は，

1) 見たい西暦，月を入力してもらう
2) 閏年をチェックし２月の日数を決定する
3) 見たい月の１日の曜日を決定する
4) 先月のPRMONルーチンを使ってカレンダーを表示する
5) 別の月を見るかどうかの確認

という順で行います。１つひとつ見ていきましょう。

●年月の入力

リスト２がここに相当します。1～9行はこのプログラムで使う定数を設定しているところです。12，13行で画面をクリアし，まずは西暦の入力からです。15，16行で画面に西暦の入力を促すメッセージを出力し，17，18行で#GETLを使って西暦を入力します。キー入力バッファはKEYBUFという名前でプログラムのいちばん後ろにくっつけてあります。#GETLルーチンはDEレジスタの内容を破壊しませんから，そのまますぐにCONVルーチンを19行で呼び出してやることができるのです。

CONVルーチンで数字を文字に変換したら20行で変換が正常に終了したかどうかをチェックします。ノンゼロなら変な文字が入っているということですから，もう一度西暦の入力をやり直します。そうでなければHLが0かどうかを判断します。これは７月のカレンダーに続けて８月のカレンダーを見たいというときに，いちいち西暦から入力し直さなくてもいいようにという配慮からです。リターンキーのみを押すとバッファの先頭に00Hが入りますから，HLは0のまま帰ってくるのです。21～23行でチェックし，HLが0のときには月の入力に飛ばします。そうでなければYEARというワークに変換した西暦をセットし，続けて月の入力に入ります。

月の入力は入力を促すメッセージが異なっていることと，12月より大きな月は受け付けないようにしているところが西暦の入力と異なっているだけです。簡単ですので追ってみてください。

このようにして西暦と月の入力が終わると以後はYEAR,MONTHの２つのワークに入っている値をもとにプログラムを進めていきます。つまり西暦もしくは月の入力を省略したときには以前入力していた値が有効になるのです。この２つのワークには初期値として１が入っています。

●２月の日数の決定

２月の日数を決定するのはカレンダーを表示するためだけではなく，各月の初日の曜日を決定するためにも必要なことです。

実際にプログラムでやることは非常に簡単なことです。入力されたYEARが閏年かどうかの判断をすればいいのですから。先月作った割り算ルーチンDIVQはHLを127以下の数で割ることしかできませんので，YEARが400で割り切れるかどうかを判断することはできません。かといって，このためだけに別の割り算ルーチンを用意するのもシャクです。そこで次のような方法を用いて解決することにしました。

1) YEARを４で割る。余りが出れば平年
2) 1)の答えを25で割る。余りが出れば閏年
3) 2)の答えを４で割る。余りが出れば平年

注目してほしいのは2)と3)です。1)で余りが出なければYEAR÷４の商をさらに25で割ります。つまりYEAR÷４÷25とするわけです。これはYEAR÷(4×25)と同じこと，つまりYEAR÷100を計算することになるのです。この結果余りが出なければさらにその商を４で割ります。つまりYEAR÷4÷25÷4です。これはYEAR÷(4×25×4)＝YEAR÷400を計算するのと同じことです。

さてプログラムのほうを見てみましょう。リスト３です。いま述べたアルゴリズムどおりに処理を進めます。まず42行でYEARをHLに取り出し，43，44行でこれを４で割ります。余りはAに入

リスト２　年月の入力

```
0000                1 ;  CALENDER PROGRAM
0000                2 ;
8000                3         ORG     8000H
8000                4         ;
8000                5 #PRINT  EQU     1FF4H
8000                6 #PRNTS  EQU     1FF1H
8000                7 #MSX    EQU     1FE5H
8000                8 #GETL   EQU     1FD3H
8000                9 #FLGET  EQU     2021H
8000               10
8000               11 SHOKI:
8000 3E 0C         12         LD      A,0CH
8002 CD F4 1F      13         CALL    #PRINT
8005               14         ;
8005 11 AC 81      15 SHKI1:  LD      DE,GETY         ; Year
8008 CD E5 1F      16         CALL    #MSX
800B 11 EF 81      17         LD      DE,KEYBUF
800E CD D3 1F      18         CALL    #GETL
8011 CD CC 80      19         CALL    CONV
8014 20 EF         20         JR      NZ,SHKI1
8016 7D            21         LD      A,L
8017 B4            22         OR      H
8018 28 03         23         JR      Z,SHKI2
801A 22 EC 81      24         LD      (YEAR),HL
801D               25         ;
801D 11 B8 81      26 SHKI2:  LD      DE,GETM         ; Month
8020 CD E5 1F      27         CALL    #MSX
8023 11 EF 81      28         LD      DE,KEYBUF
8026 CD D3 1F      29         CALL    #GETL
8029 CD CC 80      30         CALL    CONV
802C 20 EF         31         JR      NZ,SHKI2
802E 7D            32         LD      A,L
802F B7            33         OR      A
8030 28 07         34         JR      Z,URUCHK
8032 FE 0D         35         CP      12+1
8034 30 E7         36         JR      NC,SHKI2
8036 32 EE 81      37         LD      (MONTH),A
8039               38
```

リスト３　２月の日数を決定する

```
8039               39 ;  2ｶﾞﾂ ﾉ ﾆｯｽｳｦ ｹｯﾃｲｽﾙ
8039               40 ;
8039               41 URUCHK:
8039 2A EC 81      42         LD      HL,(YEAR)
803C 3E 04         43         LD      A,4
803E CD 9C 81      44         CALL    DIVQ            ; A=HL%4
8041 B7            45         OR      A
8042 20 10         46         JR      NZ,HEINEN
8044 3E 19         47         LD      A,25
8046 CD 9C 81      48         CALL    DIVQ            ; A=(HL/4)%25
8049 B7            49         OR      A               ;  (=HL%100)
804A 20 0C         50         JR      NZ,URUU
804C 3E 04         51         LD      A,4
804E CD 9C 81      52         CALL    DIVQ            ; A=((HL/4)/25)%4
8051 B7            53         OR      A               ;  (=HL%400)
8052 28 04         54         JR      Z,URUU
8054               55         ;
8054 3E 1C         56 HEINEN: LD      A,28
8056 18 02         57         JR      FEBSET
8058               58         ;
8058 3E 1D         59 URUU:   LD      A,29
805A 32 49 81      60 FEBSET: LD      (DAYS+1),A
805D               61
```

って帰ってきますから45行で0かどうかの判断をし，0でなければHEINENに飛ばして平年の処理を行います。これで1)は終了ですね。次に47，48行で先のYEAR÷4の商をさらに25で割ります。49，50行で余りのチェックをし，余りが0でなければ閏年の処理へ飛ばします。これで2)も終了です。最後に51〜54行で3)の処理をします。簡単ですから追ってみてください。

さてHEINENとURUUですが，ここではAにそれぞれの年の2月の日数をセットして，先月作ったPRMON内にある月の日数テーブルを書き換えています。以上で2月の日数を求める処理は終了です。

●1日の曜日を算出する

リスト4はいよいよカレンダープログラムのメインディッシュの登場です。プログラムの説明を読む前にもう一度「1日は何曜日？」というところを読み直しておいてください。

最初に

(m−1)＋[m年までの閏年の回数]＋[n−1月までの日数]＋1

の「(m−1)＋[m年までの閏年の回数]」を計算します。まず，65行でYEARを取り出します。これがmに相当するわけですね。66行で(m−1)を計算しスタックに保存しておきます。次に，m年までの閏年の回数を計算します。68〜70行で(m−1)÷4を計算してスタックへ保存。71〜73行で(m−1)÷100を計算してスタックへ。74，75行で(m−1)÷400を計算します。ここで，100や400で割るのには2月の日数を求めるときと同じ方法を用いています。

76行の時点でHLには(m−1)÷400が入っていますね。そこでスタックに保存してある数を取り出しながら，76〜82行で「(m−1)＋[m年までの閏年の回数]」を計算します。追ってみてください。

85〜87行は求めた数を7で割った余りを計算して，m−1年12月31日の曜日を求めているところです。「[n−1月までの日数]＋1」まで求めてから計算してもよいのですが，扱える西暦の範囲を大きくするためこのようにしました。88行でm−1年12月31日の曜日をスタックに保存しておき，まずはひと段落です。

もし，このアルゴリズムを用いずに「[n−1月までの日数]＋1」まで計算してから7で割ることにすると，最大で52745年の1月のカレンダーまで見ることが可能です。ただし，これは1月のカレンダーの場合で，12月のカレンダーは52476年までしか見ることができません。「(m−1)＋[m年までの閏年の回数]」を求めた時点で7で割ってやることにより，表示するのが何月のカレンダーであろうと52745年まで表示できるようになったわけです。

さらに，(m−1)をそのまま閏年の回数に足さず(m−1) mod 7を閏年の回数に足すようにすると，西暦65535年のカレンダーも表示させることができるようになります。ここでは処理の流れを簡単にわかりやすくするために，(m−1) mod 7のほうは使いませんでした。興味のある方は挑戦してみてください。

次に，[n−1月までの日数]を求めます。これには先月作ったPRMON内の月の日数テーブルを利用します。HLにこのテーブルの先頭アドレスをセットすると，(HL)には1月の日数が入っていますね。「INC HL」を実行すると(HL)は2月の日数になります。この調子でn−1回繰り返せば見たい年の1月からn−1月までのそれぞれの月の日数を取り出すことができますね。また，取り出すたびにその値を足し合わせていけば[n−1月までの日数]を得ることができます。

ここで困るのが「足し合わせる」という作業です。1月から11月までの日数は334日で，これはAに入れることができる範囲をオーバーしています。となると16ビットの足し算を行うしかありません。ところが16ビットの足し算で必ず使われるHLレジスタは，現在は月の日数テーブルを指すのに使用しているのです。そこで先月の最後に説明した「EX DE, HL」という命令を使うことにしましょう。「EX DE, HL」はHLとDEの内容を交換するんでしたね。ですから

```
        LD    DE, 0    ; DEに答えを入れることにする
LOOP：  LD    C, (HL)  ; いまHLはテーブルを指している
        LD    B, 0     ; BCに月の日数を取り出す
        INC   HL       ; HLを次の月に進める
        EX    DE, HL   ; DEとHLを交換
        ADD   HL, BC   ; 取り出した日数を足し込む
        EX    DE, HL   ; 再びHLはテーブルを指す
        JR    LOOP
```

というぐあいにEX命令をうまく使って，あるときはHLはテーブルを指し，あるときは日数の合計を入れるように切り換えてやることにより解決することにします。

リストを見てください。まず90行でHLが月の日数テーブルの先頭を指すようにし，91行で足し合わせた答えを入れるDEを0に初期化します。続いて92行でAにMONTH，つまり月を取り出し，計算開始です。Aをループカウンタに使います。93〜101行が月の

リスト4　1日の曜日を決定する

```
805D             62 ;  1ニチ ノ ヨウビ ヲ ケッテイスル
805D             63 ;
805D             64 CALCDAY:
805D 2A EC 81    65         LD      HL,(YEAR)
8060 2B          66         DEC     HL
8061 E5          67         PUSH    HL              ; HL=YEAR-1 (=Y0)
8062 3E 04       68         LD      A,4
8064 CD 9C 81    69         CALL    DIVQ
8067 E5          70         PUSH    HL              ; HL=Y0/4   (=Y1)
8068 3E 19       71         LD      A,25
806A CD 9C 81    72         CALL    DIVQ
806D E5          73         PUSH    HL              ; HL=Y0/100 (=Y2)
806E 3E 04       74         LD      A,4
8070 CD 9C 81    75         CALL    DIVQ            ; HL=Y0/400 (=Y3)
8073 D1          76         POP     DE
8074 B7          77         OR      A
8075 ED 52       78         SBC     HL,DE           ; HL=Y3-Y2
8077 D1          79         POP     DE
8078 19          80         ADD     HL,DE           ; HL=Y3-Y2+Y1
8079 D1          81         POP     DE
807A 19          82         ADD     HL,DE           ; HL=Y3-Y2+Y1+Y0
807B             83         ;
807B 3E 07       84         LD      A,7
807D CD 9C 81    85         CALL    DIVQ
8080 6F          86         LD      L,A
8081 26 00       87         LD      H,0
8083 E5          88         PUSH    HL            ; HL=ゼンネンノ 12ガツ 31ニチ ノ ヨウビ
8084             89         ;
8084 21 48 81    90         LD      HL,DAYS         ; table ノ セントウ adrs
8087 11 00 00    91         LD      DE,0            ; Acc
808A 3A EE 81    92         LD      A,(MONTH)       ; loop counter
808D 3D          93 CLCDY1: DEC     A
808E 28 09       94         JR      Z,CLCDY2        ; loop end
8090 4E          95         LD      C,(HL)
8091 06 00       96         LD      B,0             ; BC=ツキ ノ ニッスウ
8093 23          97         INC     HL
8094 EB          98         EX      DE,HL           ; HL=ヨウビ
8095 09          99         ADD     HL,BC
8096 EB         100         EX      DE,HL           ; HL=pointer
8097 18 F4      101         JR      CLCDY1
8099            102         ;
8099 E1         103 CLCDY2: POP     HL            ; HL=ゼンネンノ 12ガツ 31ニチ ノ ヨウビ
809A 19         104         ADD     HL,DE
809B 23         105         INC     HL
809C 3E 07      106         LD      A,7
809E CD 9C 81   107         CALL    DIVQ
80A1 6F         108         LD      L,A
80A2 26 00      109         LD      H,0
80A4 E5         110         PUSH    HL              ; HL=1ニチ ノ ヨウビ
80A5            111
```

リスト5　カレンダーの表示

```
80A5            112 ;  カレンダーノ ヒョウジ
80A5            113 ;
80A5 3E 0C      114         LD      A,0CH
80A7 CD F4 1F   115         CALL    #PRINT
80AA 2A EC 81   116         LD      HL,(YEAR)
80AD CD E9 80   117         CALL    PRYEAR
80B0 11 C5 81   118         LD      DE,MES
80B3 CD E5 1F   119         CALL    #MSX
80B6            120         ;
80B6 E1         121         POP     HL              ; HL=センゲツ マツジツ ノ ヨウビ
80B7 3A EE 81   122         LD      A,(MONTH)
80BA CD FF 80   123         CALL    PRMON
80BD            124         ;
80BD 11 D2 81   125         LD      DE,KEYIN
80C0 CD E5 1F   126         CALL    #MSX
80C3 CD 21 20   127         CALL    #FLGET
80C6 FE 0D      128         CP      0DH             ; CR
80C8 CA 00 80   129         JP      Z,SHOKI
80CB C9         130         RET
80CC            131
```

日数を足し合わせるループです。93，94行でAを1減じ，ループ終了かどうかを判断します。もし1月のカレンダーを表示するのならここで引っ掛かりますからループを一度も回らずに終了するわけです。95～101行は先ほど説明したHLとDEをひっくり返しながらの日数の合計計算です。

こうしてループを終了すると103行にきます。いまDEには表示する年の1月から表示月までの日数が入っています。また，スタックには表示する前年の12月31日の曜日が保存してありますから，103，104行でこの2つを足し合わせ「(m−1)＋[m年までの閏年の回数]＋[n−1月までの日数]」までの計算が終了です。これに105行で1を足し，106，107行で7で割った余りを求めます。この余りを108，109行でHLにセットすれば，HLには表示したい月が何曜日から始まっているのかが求まります。

これで先月作ったPRMONを呼び出す用意はすべて完了しました。このまま表示するだけでもよいのですが，どうせなら表示しているカレンダーが西暦何年のものなのかという情報も画面に出すことにしましょう。そこでHLを110行で保存しておきます。

●カレンダーを表示する

実際にどのようにカレンダーを表示するのかはリスト5を見ながら説明していきましょう。

画面には入力された西暦と月が残っていますから，114，115行で画面をクリアします。次に116，117行で表示する西暦をYEARから取り出し，PRYEARというサブルーチンを呼び出して画面に10進数で表示します。PRYEARがどのようになっているのかはまたあとでお話ししましょう。10進表示した西暦に続いて，118，119行で「年のカレンダー」と表示します。これで画面の最上段には「○○年のカレンダー」と表示されたわけです。

ここで，スタックに保存しておいた1日の曜日をHLへ取り出し，表示する月をAに取り出してPRMONを呼び出します(121～123行)。そう，PRMONはHLに1日の曜日，Aに表示する月を入れて呼び出すんでしたね。

カレンダー表示が終わると125～127行で「HIT <CR> to continue」と表示し，#FLGETルーチンを使ってカーソル点滅1文字入力を行います。これは別の月のカレンダーを見たいと思ったときに簡単に見ることができるようにという配慮です。#FLGETルーチンからはAに押されたキーのアスキーコードを入れて帰ってきます。そこで128行で押されたのがリターンキーだったのかどうかという判定を行い，押されたのがリターンキーだったのなら129行でプログラムの先頭に飛んで再び西暦と月の入力を行います。リターンキーでなかった場合は130行でプログラムは終了です。

再帰を使って西暦表示

リスト6が西暦を10進数で表示するサブルーチンPRYEARです。このサブルーチンはちょっと変わった作り方をしてありますので，初心者の方には難しいかもしれません。読み飛ばしてもらってもけっこうです。

これまで何度か取り上げた10進表示ルーチンとは明らかに違う形をしているのにお気付きでしょうか。これまでの10進表示ルーチンは内部にワークを持っており，頭に付く不要な0をこのワークを調べることによって排除していました。なによりおかしいのは178行にある「CALL PRYEAR」という命令です。自分自身を呼び出しているのです。いったいどうしたことでしょう。

じつはPRYEARは再帰(リカーシブコール)という方法を用いてプログラムしてあるのです。LISP，C，PASCALなどですっかり有名な再帰プログラムですが，ご存じでない方もいらっしゃるでしょうから簡単に説明しておきましょう。

再帰とは手続きや関数が自分自身の内部で再び自分自身を呼び出すことをいいます。たとえば0からnまでの合計を求める関数SUMを考えてみましょう。SUM(n)は次のように定義されます。

1)　もしnが0なら答えは0

2)　そうでなければ答えはSUM(n−1)にnを加えたもの

たとえば0から3までの合計はこの定義によると，0から2までの合計に3を加えたものだということになります。

確かにそのとおりだけど，実際にそんな計算ができるのだろうかという疑問が必ずわくと思います。再帰計算のできる言語はよく工夫されていて，次のように解決します。例としてSUM(3)をやってみましょう。SUM(3)は次のように展開されます。

　SUM(3)＝SUM(2)＋3

ここでSUM(2)はまだ値が求まっていません。そこで

　SUM(2)＝SUM(1)＋2

としてSUM(2)を求めにいきます。ところがSUM(1)もまた求まっていません。

　SUM(1)＝SUM(0)＋1

SUM(0)は定義してあり，答えは0ですね。ですから

　SUM(1)＝0＋1

となってSUM(1)が求まります。SUM(1)が求まれば

　SUM(2)＝1＋2

となってSUM(2)も求まり，

　SUM(3)＝3＋3

でSUM(3)の答えを求めることができるのです。

LISPなど再帰の使えるインタプリタではこのような方法で(正確にはこのとおりではない)計算を進めます。ではCやPASCALなどのコンパイラ型言語ではどう解決しているのかといいますと，これはコンパイラの作者がどのように処理しようと考えるかによって違ってくるのですが，一般的には関数の引数をスタック上に取ることにより実現しています。詳しいことはそのうち改めてやることにして，どのように再帰プログラムを作るかという方向で進めていきましょう。

慣れてしまうとなんでもない再帰なのですが，最初のうちはどうやって再帰的に表現したらいいのかと悩むところです。コツは

リスト6　西暦の表示

```
80E9            164 ;  Print Out Year
80E9            165 ;
80E9            166 ;  in   : HL = YEAR
80E9            167 ;
80E9            168 PRYEAR:
80E9 3E 0A      169         LD      A,10
80EB CD 9C 81   170         CALL    DIVQ
80EE 47         171         LD      B,A
80EF 7D         172         LD      A,L
80F0 B4         173         OR      H
80F1 78         174         LD      A,B
80F2 28 05      175         JR      Z,PRYR1         ; ｺﾀｴ ﾊ 0 ?
80F4            176         ;
80F4 F5         177         PUSH    AF              ; ｱﾏﾘ ｦ ﾎｿﾞﾝ
80F5 CD E9 80   178         CALL    PRYEAR
80F8 F1         179         POP     AF
80F9 C6 30      180 PRYR1:  ADD     A,'0'
80FB CD F4 1F   181         CALL    #PRINT
80FE C9         182         RET
80FF            183
```

「今から作ろうとする関数がすでにできていると思い込む」ことです。西暦を表示する関数pryearをX68000のX-BASICで書いてみましょう。

```
func pryear(year)
  if year/10=0 then {
    print chr$(&H30+(year mod 10));
  } else {
   pryear(year/10)
    print chr$(&H30+(year mod 10));
  }
endfunc
```

ここでpryearは次のように定義されています。

1) もしyearを10で割った答えが0だったら，その余りを画面に表示する

2) そうでなかったら，year÷10をpryearで表示させて，その後ろにyearを10で割った余りを表示する

注目してほしいのは2)です。たとえば1234を表示させようとするならば，pryearによって123を表示させ，その後ろに4を表示すればいいということなのです。pryearをいま定義しているにもかかわらずこのようにすでにできあがったものとして利用する。これが大切な考え方なのです。

もっともこれは考え方であって，定義が終わったら頭の中で追いかけてみて，ちゃんと動くことを確かめなければいけません。pryear(123)を追いかけてみますと，

1) pryear(123)

2) ⇨pryear(12):print "3"

3) ⇨pryear(1):print "2"::print "3"

4) ⇨print "1";:print "2";:print "3"

となりますから，確かに画面には123と表示されることが確認できました。

ではpryearのマシン語版を見ることにしましょう。PRYEARではHLに表示したい数をセットしてから呼び出すことにしました。まず169，170行でHLを10で割り，171行で余りをBに保存しておきます。172，173行で割り算の商HLが0かどうかを調べ，Bに保存しておいた余りを174行でAに戻しておきます。そして，商が0だった場合には余りを表示するために180行へ飛ばします(175行)。商が0でなかった場合には177行にきます。ここで余りをスタックに保存しておいて，10で割った商を持って再びPRYEARを呼び出すのです。

PRYEARでHL÷10を表示させたあとは179行です。保存しておいた余りを取り出し，180行で数値を数字に直して182行で表示すれば終了です。

このように，マシン語で再帰を実現しようとする場合も考え方は同じです。PRYEARができているものとしてプログラムすればいいのです。ただし，ひとつだけ気をつけなければならないことがあります。それはマシン語には引数もローカル変数もないということです。177，179行で余りを保存しなければならないのはこのためです。さもないと，再帰したときの割り算によってAが壊されてしまい，再帰から帰ってきたときに余りを表示してやることができなくなってしまいます。turboBASICで再帰をするのと同じようなものだと考えればいいでしょう。

再帰はアルゴリズムを簡潔に表現できる非常に強力な方法です。しかし，スタックを異常に消費し，実行は速くないというマイナスの面も持ち合わせています。再帰で書くかループで書くかとい

大ちゃんのワンポイントレッスン

私は以前マシン語体操で作ったメモリエディタを使っているのですが，16進数を入力するのが面倒でしかたありません。そこで，テンキーを16進キーにする方法を教えてほしいのです。入力が全部テンキーでできれば非常に楽になります。ぜひとも取り上げてください。
三重県　水谷　良彦

テンキーを16進キーにするというのは1986年1月号の質問箱で取り上げられています。質問箱の方法はBIOSやIOCSを書き換えて，カーソル点滅1文字入力だろうが，1行入力だろうが，はたまたリアルタイムキー入力だろうが，テンキーが16進キーになるという非常においしい方法です。X1ではテンキーの演算記号を，MZ-2000ではシフトキーを押しながら0～6のキーを押せばA～Fが入力できるようになっています。この方法はシステム内部を書き換えてしまうため，システムに対する深い知識がないと危険です。1カ所書き換えた影響が，思いもよらないところに現れることがあるからです。そこで，#FLGETルーチンを使ってテンキーを16進キーにする方法をここでは説明しましょう。

テンキーを16進キーにするには，#FLGETで得たアスキーコードが演算記号だったらそれに対応するA～Fのアスキーコードに変換してやればいいだけです。つまり，#FLGETから帰ってきたら

```
CP '/'
```

として演算記号の'/'が入力されたかどうかを調べます。もしそうなら

```
LD A,'A'
```

とやって#FLGETによって取り込まれた「押されたキーのアスキーコード」を変えてやればいいのです。具体的には，

1) メモリエディタのリストの最初でラベル定義されている「#FLGET EQU 2021H」という行を削除する

2) 以下に示す#FLGETをプログラムの最後に付け加える

というぐあいにします。新しい#FLGETは次のようなプログラムです。

```
FLGET    EQU  2021H

#FLGET:
        CALL FLGET    ; カーソル点滅1文字入力の呼び出し
        PUSH BC       ; Bを壊すので保存しておく
;
        LD   B,'A'    ; Bは'A'のアスキーコード
        CP   '/'      ; 入力されたのは'/'か
        JR   Z,FOUT   ; そうなら終了
;
        INC  B        ; Bは'B'のアスキーコード
        CP   '*'      ; 入力されたのは'*'か
        JR   Z,FOUT   ; そうなら終了
;
        ......        ('C'～'E'の処理)
;
        INC  B        ; Bは'F'のアスキーコード
        CP   '.'      ; 入力されたのは'.'か
        JR   NZ,FOUT1 ; そうでなければFOUT1へ
;
FOUT:   LD   A,B      ; 対応するアスキーコードをAへ入れる
FOUT1:  POP  BC       ; 保存しておいたBを取り出す
        RET
```

こうしたあとでアセンブルしなおせば，プログラムの他の部分に手を加えることなく簡単に16進対応のキー配列にすることができます。この方法の欠点はテンキーの演算記号だけではなくメインキーの演算記号を押してもA～Fに変換されてしまうことですが，マシン語入力ツールのようなものなら問題ないでしょう。当然のことですが，#FLGETを使っていないツールではこの方法は使えません。

また，テンキーのない機種でもメインキーの一部をこの方法で16進キーにしてしまうことができます。メインキーの7～9を生かしてそのあたりにうまく16個のキーを割り振るのもいいでしょう。とにかく自分の使いやすいようにキーボードをアレンジできるというのが魅力です。

最後に私のマシン語入力法を紹介しておきましょう。私は右手でテンキー，左手でメインキーを受け持ち，0～9なら右手で，A～Fなら左手で入力しています。こいつは速いですよ。それではまた来月。

うのはかなり好みに左右されるでしょう。ただ，ループを使って書けるものはすべて再帰で書くことができるが，再帰で書くことのできるものの中にはループでは書けないものがあるそうです。皆さんはどちらがお好みでしょう。私ですか？　私は再帰大好き人間です。

来月は計算ルーチンをまとめてやっつけようと考えています。Z80の命令には用意されていない掛け算・割り算のための計算ルーチンを，0〜65535という通常扱っている範囲を越えた数を計算するにはどうすればいいのかという題目をまじえて紹介していきます。ご期待ください。

リスト7　先月のサブルーチンとメッセージデータ

```
80FF              184 ;  PRINT MONTH
80FF              185 ;
80FF              186 ;  in   :  A = Month to be printed
80FF              187 ;       : HL = First Day ( 0,1,2,..)
80FF              188 ;
80FF              189 ;  brkn : AF,BC,HL
80FF              190 ;
80FF              191 PRMON:
80FF D5           192         PUSH    DE
8100              193         ;
8100 F5           194         PUSH    AF
8101 CD 7E 81     195         CALL    PRTDT1
8104 3E 29        196         LD      A,')'
8106 CD F4 1F     197         CALL    #PRINT
8109 F1           198         POP     AF
810A              199         ;
810A 11 54 81     200         LD      DE,WEEK
810D CD E5 1F     201         CALL    #MSX
8110              202         ;
8110 E5           203         PUSH    HL
8111 21 48 81     204         LD      HL,DAYS          ; Top of Table
8114 3D           205         DEC     A                ; 0=1ｶﾞﾂ, 1=2ｶﾞﾂ, ...
8115 5F           206         LD      E,A
8116 16 00        207         LD      D,0              ; DE=Month
8118 19           208         ADD     HL,DE
8119 4E           209         LD      C,(HL)           ; C=Days of the Month
811A E1           210         POP     HL
811B              211         ;
811B 41           212         LD      B,C
811C 0C           213         INC     C
811D 7D           214         LD      A,L
811E B4           215         OR      H
811F E5           216         PUSH    HL
8120 28 14        217         JR      Z,PRMON2
8122              218         ;
8122 11 73 81     219         LD      DE,TAB1          ; ﾀﾞｲ 1ｼｭｳﾉ
8125 CD E5 1F     220 PRMON0: CALL    #MSX             ;  ｽﾍﾟｰｽ ｦ ｼｭﾂﾘｮｸ
8128 2D           221         DEC     L
8129 20 FA        222         JR      NZ,PRMON0
812B E1           223         POP     HL
812C              224         ;
812C E5           225 PRMON1: PUSH    HL
812D 3E 07        226         LD      A,7
812F CD 9C 81     227         CALL    DIVQ
8132 B7           228         OR      A
8133 CC 41 81     229         CALL    Z,CRTAB
8136 79           230 PRMON2: LD      A,C
8137 90           231         SUB     B
8138 CD 78 81     232         CALL    PRTDT            ; Display the DATE
813B E1           233         POP     HL
813C 23           234         INC     HL
813D 10 ED        235         DJNZ    PRMON1
813F              236         ;
813F D1           237         POP     DE
8140 C9           238         RET
8141              239 ;
8141              240 CRTAB:                           ; print CR and TAB
8141 11 72 81     241         LD      DE,TAB
8144 CD E5 1F     242         CALL    #MSX
8147 C9           243         RET
8148              244 ;
8148 1F 1C 1F 1E  245 DAYS:   DEFB    31,28,31,30
814C 1F 1E 1F 1F  246         DEFB    31,30,31,31
8150 1E 1F 1E 1F  247         DEFB    30,31,30,31
8154              248 ;
8154 20 20 53 75  249 WEEK:   DFFM    "  Sun Mon Tue Wed Thu Fri Sat"
8158 6E 20 4D 6F
815C 6E 20 54 75
8160 65 20 57 65
8164 64 20 54 68
8168 75 20 46 72
816C 69 20 53 61
8170 74
8171 0D           250         DEFB    0DH
8172 0D           251 TAB:    DEFB    0DH
8173 20 20 20 20  252 TAB1:   DEFM    "    "
8177 00           253         DEFB    0
8178              254
8178              255 ;  PRINT DATE
8178              256 ;
8178              257 ;  in   : A = DATE to be printed
8178              258 ;
8178              259 ;  brkn : AF
8178              260 ;
8178              261 PRTDT:
8178 CD F1 1F     262         CALL    #PRNTS
817B CD F1 1F     263         CALL    #PRNTS
817E E5           264 PRTDT1: PUSH    HL
817F 6F           265         LD      L,A
8180 26 00        266         LD      H,0              ; HL = DATE
8182 3E 0A        267         LD      A,10
8184 CD 9C 81     268         CALL    DIVQ             ; DATE/10
8187 F5           269         PUSH    AF               ; 1ﾉ ｸﾗｲ ｦ save
8188 7D           270         LD      A,L
8189 C6 30        271         ADD     A,'0'
818B FE 30        272         CP      '0'              ; 10ﾉ ｸﾗｲ ﾊ 0 ?
818D 20 02        273         JR      NZ,PRTDT2
818F 3E 20        274         LD      A,' '
8191 CD F4 1F     275 PRTDT2: CALL    #PRINT
8194 F1           276         POP     AF               ; 1ﾉ ｸﾗｲ ｦ get
8195 C6 30        277         ADD     A,'0'
8197 CD F4 1F     278         CALL    #PRINT
819A E1           279         POP     HL
819B C9           280         RET
819C              281
819C              282 ;  QUOTIENT ROUTINE
819C              283 ;
819C              284 ;  in   : HL = Num
819C              285 ;       :  A = divisor ( < 128 )
819C              286 ;
819C              287 ;  out  : HL = HL / A
819C              288 ;       :  A = HL % A
819C              289 ;
819C              290 DIVQ:
819C C5           291         PUSH    BC
819D 4F           292         LD      C,A              ; Copy Divisor
819E AF           293         XOR     A
819F              294         ;
819F 06 10        295         LD      B,16             ; Loop Counter
81A1 29           296 DIVQ1:  ADD     HL,HL            ; Shift Left HL
81A2 8F           297         ADC     A,A              ; Shift Left A with CY
81A3 B9           298         CP      C
81A4 38 02        299         JR      C,DIVQ2
81A6 91           300         SUB     C
81A7 23           301         INC     HL               ; Set ANS bit
81A8 10 F7        302 DIVQ2:  DJNZ    DIVQ1
81AA              303         ;
81AA C1           304         POP     BC
81AB C9           305         RET
81AC              306
81AC              307 ;  Messages
81AC              308 ;
81AC 49 6E 70 75  309 GETY:   DEFM    "Input Year"
81B0 74 20 59 65
81B4 61 72
81B6 0D 00        310         DEFB    0DH,0
81B8 49 6E 70 75  311 GETM:   DEFM    "Input Month"
81BC 74 20 4D 6F
81C0 6E 74 68
81C3 0D 00        312         DEFB    0DH,0
81C5 C8 DD C9 20  313 MES:    DEFM    "ﾈﾝﾉ ｶﾚﾝﾀﾞｰ"
81C9 B6 DA DD C0
81CD DE 2D
81CF 0D 0D 00     314         DEFB    0DH,0DH,0
81D2 0D 0D 0D     315 KEYIN:  DEFB    0DH,0DH,0DH
81D5 48 49 54 20  316         DEFM    "HIT <CR> to continue  "
81D9 3C 43 52 3E
81DD 20 74 6F 20
81E1 63 6F 6E 74
81E5 69 6E 75 65
81E9 20 20
81EB 00           317         DEFB    0
81EC              318 ;  Works
81EC              319 ;
81EC 01 00        320 YEAR:   DEFW    1
81EE 01           321 MONTH:  DEFB    1
81EF              322 ;
81EF 00 00 00 00  323 KEYBUF: DEFS    256
```

リスト8　万年暦プログラム全ダンプリスト

```
8000 3E 0C CD F4 1F 11 AC 81 : 68
8008 CD E5 1F 11 EF 81 CD D3 : F2
8010 1F CD CC 80 20 EF 7D B4 : 78
8018 28 03 22 EC 81 11 B8 81 : 04
8020 CD E5 1F 11 EF 81 CD D3 : F2
8028 1F CD CC 80 20 EF 7D B7 : 7B
8030 28 07 FE 0D 30 E7 32 EE : 71
8038 81 2A EC 81 3E 04 CD 9C : C3
8040 81 B7 20 10 3E 19 CD 9C : 28
8048 81 B7 20 0C 3E 04 CD 9C : 0F
8050 81 B7 28 04 3E 1C 18 02 : D8
8058 3E 1D 32 49 81 2A EC 81 : EE
8060 2B E5 3E 04 CD 9C 81 E5 : 21
8068 3E 19 CD 9C 81 E5 3E 04 : 68
8070 CD 9C 81 D1 B7 ED 52 D1 : 82
8078 19 D1 19 3E 07 CD 9C 81 : 32
---------------------------------
SUM: F7 51 EE A8 73 8B 42 93 CFF0

8080 6F 26 00 E5 21 48 81 11 : 75
8088 00 00 3A EE 81 3D 28 09 : 17
8090 4E 06 00 23 EB 09 EB 18 : 6E
8098 F4 E1 19 23 3E 07 CD 9C : BF
80A0 81 6F 26 00 E5 3E 0C CD : 12
80A8 F4 1F 2A EC 81 CD E9 80 : E0
80B0 11 C5 81 CD E5 1F E1 3A : 43
80B8 EE 81 CD FF 80 11 D2 81 : 1F
80C0 CD E5 1F CD 21 20 FE 0D : EA
80C8 CA 00 80 C9 21 00 00 1A : 4E
80D0 13 FE 30 38 12 FE 3A 30 : F3
80D8 0E 29 4D 44 29 29 09 D6 : F9
80E0 30 4F 06 00 09 18 E8 B7 : 45
80E8 C9 3E 0A CD 9C 81 47 7D : BF
80F0 B4 78 28 05 F5 CD E9 80 : 84
80F8 F1 C6 30 CD F4 1F C9 D5 : 65
---------------------------------
SUM: 7B B8 75 82 A1 9C 2B 8C B009

8100 F5 CD 7E 81 3E 29 CD F4 : E9
8108 1F F1 11 54 81 CD E5 1F : C7
8110 E5 21 48 81 3D 5F 16 00 : 81
8118 19 4E E1 41 0C 7D B4 E5 : AB
8120 28 14 11 73 81 CD E5 1F : 12
8128 2D 20 FA E1 E5 3E 07 CD : 1F
8130 9C 81 B7 CC 41 81 79 90 : 6B
8138 CD 78 81 E1 23 10 ED D1 : 98
8140 C9 11 72 81 CD E5 1F C9 : 67
8148 1F 1C 1F 1E 1F 1E 1F 1F : F3
8150 1E 1F 1E 1F 20 20 53 75 : 82
8158 6E 20 4D 6F 6E 20 54 75 : A1
8160 65 20 57 65 64 20 54 68 : 81
8168 75 20 46 72 69 20 53 61 : 8A
8170 74 0D 0D 20 20 20 20 00 : 0E
8178 CD F1 1F CD F1 1F E5 6F : 0E
---------------------------------
SUM: 5F 04 C0 89 2A 30 5F 4F E719

8180 26 00 3E 0A CD 9C 81 F5 : 4D
8188 7D C6 30 FE 30 20 02 3E : 01
8190 20 CD F4 1F F1 C6 30 CD : B4
8198 F4 1F E1 C9 C5 4F AF 06 : 86
81A0 10 29 8F B9 38 02 91 23 : 6F
81A8 10 F7 C1 C9 49 6E 70 75 : 2D
81B0 74 20 59 65 61 72 0D 00 : 32
81B8 49 6E 70 75 74 20 4D 6F : EC
81C0 6E 74 68 0D 00 C8 DD C9 : C5
81C8 20 B6 DA DD C0 DE 2D 0D : 65
81D0 0D 00 0D 0D 0D 48 49 54 : 19
81D8 20 3C 43 52 3E 20 74 6F : 32
81E0 20 63 6F 6E 74 69 6E 75 : 20
81E8 65 20 20 00 01 00 01 00 : A7
---------------------------------
SUM: D4 49 7D 03 89 4A F3 1B DC58
```

BASICで数学と遊ぶ　第4回

一次変換と二次曲線の標準形

高校数学のなかでも一次変換による図形の移動はなかなか刺激的なテーマです。また，今月は二次曲線の標準形についても考えてみたいと思います。一見複雑な曲線の方程式も，実は標準的な曲線を移動させたり回転させたりしたものであることがわかるでしょう。

Yaso Tsutomu
八十　勉

1 一次変換による図形の移動

前回は関数のグラフを描くプログラムをいろいろと考えてみましたが，今回はまず種々のグラフ（図形）を一次変換によって移動させてみましょう。

一次変換fは$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$のような行列で表すことができ，これによって点（$x, y$）が点（$x', y'$）に移されるとすると，

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax+by \\ cx+dy \end{pmatrix}$$

すなわち，

$$x'=ax+by \qquad y'=cx+dy$$

となります。

図形は点の集合と考えられますから，曲線cがこの一次変換fでc'に移ったとすると，c'のグラフを描くには曲線c上の点pのfによる像p'を描けばよいことになります。これは特に難しい計算もなくむしろ簡単です。

ではさっそくプログラムを紹介しましょう。リスト1は一次変換によって図形がどのような図形に移るかを調べるものです。扱っている関数は，

1)　陽関数　$y=f(x)$
2)　極方程式　$r=f(s)$
3)　媒介変数表示　$x=f(x)$，$y=g(t)$
4)　一般の二次曲線

となっており，最初に番号を選択します。

1)～3)のグラフの描き方についてはすでに前回でやったところですが，4)についてはこのあとの二次曲線の標準化のところで説明をします。

基本的には，どの場合についても，パラメータを変化させて曲線上にある点の座標X，Yを求めてからサブルーチン2930行へ飛び，点（X，Y)が一次変換により点(XX，YY)に移ることを計算します。そして2960行で元の曲線を描き，2970行で移動後の曲線を描くというようになっています。

1340～1370行で行列の要素をINPUTしますが，HuBASICにはCALCという命令があるので，行列の要素に直接数をINPUTするのでなく文字変数 M$(1,1) に cos(π/6) などをINPUTし，A(1,1)＝CALC(M$(1，1)）として求めています。これだと，COS（π/6)≒0.87を計算してINPUTするといった手間が省けて非常に便利です。

リストと共に実行例を示しておきます。

X1turboは400ラインで

ここで紹介するプログラムはいずれもX1のBASIC CZ-8FB01/8CB01 用となっていますので，グラフィックはすべて縦200ラインで表示されるようになっています。X1turboではプログラムを少し変更すればより緻密に表示できる400ラインで使うことができるようになります。200ラインではどうしてもグラフが粗くなりがちです。turboユーザーの方はぜひ400ラインでグラフを表示させてみてください。

〈変更方法〉

まずプログラムの初期設定のところで

```
INIT：OPTIONSCREEN 0
WIDTH 80, 25, 1, 0：KMODE 0
KLIST 0：CONSOLE 0, 25
```

としておきます。そしてリスト1の1430行，リスト2の1130行を

```
WINDOW(…, 199)→WINDOW(…, 399)
```

に変えるなど，グラフィック関係の命令のY座標をすべて400ライン用に変更すれば終了です。

2 二次曲線の標準化

直線gと交わる直線ℓを，gを軸としてgのまわりに回転すると円錐が得られますが，この円錐を平面αで切ったときの切り口の曲線が楕円，双曲線，放物線となることが古くから知られています。これらの曲線は円錐曲線と呼ばれてきました(図1)。

デカルトが座標の考えを導入して以来，図形の研究（幾何学）と式の計算(代数学)とが結び付き，新しい数学の分野が開かれました（解析幾何学)。そして，すべての曲線がひとつの方程式で表されるようになりました。直線は一次方程式，円錐曲線はすべて二次方程式

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1，\frac{x^2}{a^2}-\frac{y^2}{b^2}=1，\quad y^2=4px$$

で表されることがわかりました。また逆に，一般の二次方程式

$$ax^2+bxy+cy^2+dx+ey+f=0$$

図1　円錐曲線

の表す曲線がやはり楕円，双曲線，放物線になることもわかっています。ここでは，このことについて考えてみようと思います。

ひとつのやり方としては，この方程式を y の方程式

$$cy^2+(bx+e)y+ax^2+dx+f=0$$

と考え，二次方程式の解の公式を用いて解き，y を x の陽関数で表します。すなわち，

$$y=\frac{-(bx+e)\pm\sqrt{(bx+e)^2-4c(ax^2+dx+f)}}{2c}$$

とするわけです。

このように書けば簡単なようですが，係数の値によって場合分けをしなければならないのでかなり面倒です。

$$\begin{cases} c\neq0\text{のとき} \\ \quad y=\dfrac{-(bx+e)\pm\sqrt{\mathrm{D}(x)}}{2c} \\ \qquad :\mathrm{D}(x)=(bx+e)^2-4c(ax^2+dx+f) \\ c=0\text{のとき}\Rightarrow \end{cases}$$

$$\Rightarrow\begin{cases} b\neq0\text{のとき} \\ \quad y=-\dfrac{ax^2+dx+f}{bx+e} \\ \quad a\left(-\dfrac{e}{b}\right)^2+d\left(-\dfrac{e}{b}\right)+f=0\text{なら}\quad x=-\dfrac{e}{b} \\ b=0\text{のとき}\Rightarrow \end{cases}$$

$$\Rightarrow\begin{cases} e\neq0\quad y=-\dfrac{ax^2+dx+f}{e} \\ e=0\text{のとき}\Rightarrow \end{cases}$$

$$\Rightarrow\begin{cases} a\neq0\quad x=\dfrac{-d\pm\sqrt{d^2-4af}}{2a} \\ a=0\text{のとき}\Rightarrow \end{cases}$$

$$\Rightarrow\begin{cases} d\neq0\quad x=-\dfrac{f}{a} \\ d=0\quad\text{なし} \end{cases}$$

この方法はグラフを描くことはできますが，式の形がさまざまで円錐曲線との関係もよくわかりません。

そこで，行列による一次変換を使った主軸変換という方法を用いることにします。この方法は高校数学の程度を越えていますが（線型代数学，解析幾何学），原理的には代数幾何で出てくる一次変換の回転移動を用いているだけなので難しいものではありません。ただ一般的に扱うのは大変なので具体的な例で考えてみましょう。

〈例1〉 $c_1:2x^2+4xy-y^2-20x-8y+32=0$

$$2x^2+4xy-y^2-20x-8y+32$$
$$=2(x-x_0)^2+4(x-x_0)(y-y_0)-(y-y_0)^2+c$$

とおいて，x, y の係数を比較すると，

$$-4x_0-4y_0=-20 \qquad (1)$$
$$-4x_0+2y_0=-8 \qquad (2)$$
$$2x_0{}^2+4x_0y_0-y_0{}^2+c=32 \qquad (3)$$

(1), (2) より

$$x_0=3,\quad y_0=2$$

(3) に代入して

$$c=-6$$

よって c_1 は，

$$2(x-3)^2+4(x-3)(y-2)-(y-2)^2=6$$

と変形され，これは c_1 を x 軸方向に -3，y 軸方向に -2 だけ平行移動すると，

$$c_2:2x^2+4xy-y^2=6 \quad (4)$$

という方程式で表される曲線となることを示しています。

次に c_2 を原点のまわりに $-\theta$ だけ回転させた曲線を c_3 とし，c_2 上の点 $p(x, y)$ が c_3 上の点 $p'(x', y')$ に移ったとすると，

$$\begin{pmatrix}x'\\y'\end{pmatrix}=\begin{pmatrix}\cos(-\theta) & -\sin(-\theta)\\ \sin(-\theta) & \cos(-\theta)\end{pmatrix}\begin{pmatrix}x\\y\end{pmatrix}$$

$$\therefore\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}\cos(-\theta) & -\sin(-\theta)\\ \sin(-\theta) & \cos(-\theta)\end{pmatrix}^{-1}\begin{pmatrix}x'\\y'\end{pmatrix}$$

$$=\begin{pmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{pmatrix}\begin{pmatrix}x'\\y'\end{pmatrix}$$

$$\therefore\begin{cases}x=x'\cos\theta-y'\sin\theta\\ y=x'\sin\theta+y'\cos\theta\end{cases}$$

これを(4)に代入すると c_2 は，

$$2(x'\cos\theta-y'\sin\theta)^2+4(x'\cos\theta-y'\sin\theta)$$
$$\cdot(x'\sin\theta+y'\cos\theta)-(x'\sin\theta+y'\cos\theta)^2$$
$$=6$$

これを x', y' について整理すると，

$$(2\cos^2\theta+4\sin\theta\cos\theta-\sin^2\theta)x'^2+(-4\sin\theta$$
$$\cos\theta+4\cos^2\theta-4\sin^2\theta-2\sin\theta\cos\theta)x'y'$$
$$+(2\sin^2\theta-4\sin\theta\cos\theta-\cos^2\theta)y'^2=6$$

となり，$x'y'$ の係数を 0 とすると，

$$-6\sin\theta\cos\theta+4\cos^2\theta-4\sin^2\theta=0$$
$$\therefore 4\cos2\theta=3\sin2\theta$$
$$\therefore \tan2\theta=\frac{4}{3}$$

となります。

さて，BASICにはATN(X)という関数があり，tan の値がXである角の値を返してきます。

2θ＝ATN(4/3)＝0.92729522ラジアンで，θ＝0.46364761ラジアンとなります。

この値を x'^2 の係数，y'^2 の係数に代入して計算させると，

$$2\cos^2\theta+4\sin\theta\cos\theta-\sin^2\theta=3$$
$$2\sin^2\theta-4\sin\theta\cos\theta-\cos^2\theta=-2$$

となります。

c_2 を原点のまわりに−0.4636471ラジアンだけ回転させると，

$$c_3:3x^2-2y^2=6$$

になることを示しています。これは，

$$\frac{x^2}{(\sqrt{2})^2}-\frac{y^2}{(\sqrt{3})^2}=1$$

で，つまり双曲線ということです。

この c_3 上の点を原点のまわりに0.4636761ラジアン回転し，その点を x 軸方向に3，y 軸方向に2だけ平行移動した点は元の曲線 c_1 上にあるはずです。このようにして，与えられた曲線を描くことができるわけです。

〈例2〉 $x^2-2xy+y^2+4y-11=0$

例1と同様に，

$$x^2-2xy+y^2+4y-11$$
$$=(x-x_0)^2-2(x-x_0)(y-y_0)+(y-y_0)^2+c$$

とおくと，

$$-2x_0+2y_0=0$$
$$+2x_0-2y_0=4$$

となり解はありません。

一次の項を消去することはできませんが，与えられた式をよく見ると，

$$(x-y)^2+4y-11=0$$

と変形できることがわかります。そこで，

$$x'=\frac{-x-y}{\sqrt{2}}$$

$$y'=\frac{x-y}{\sqrt{2}}$$

と変換すると，

$$y=\frac{x'+y'}{-\sqrt{2}}$$

となるので，

$$2y'^2-2\sqrt{2}(x'+y')-11=0$$
$$y'^2-\sqrt{2}x'-\sqrt{2}y'-\frac{11}{2}=0$$
$$\left(y'-\frac{\sqrt{2}}{2}\right)^2=\sqrt{2}(x'+3\sqrt{2})$$

となり，頂点が原点へくるように平行移動すると，

$$y''^2=\sqrt{2}x''$$

という放物線になります。

この場合は，まず原点のまわりに135°回転し，ついで平行移動すると標準形になります。グラフを描くには，まず $y^2=\sqrt{2}x$ 上の点を考え，その点を x 軸方向に $-3\sqrt{3}$，y 軸方向に $+\sqrt{2}/2$ だけ平行移動し，さらに原点のまわりに−135°回転すると与えられた方程式を満たす点が求まります。

このように考えると，二次曲線，

$$ax^2+2hxy+by^2+gx+fy+c=0$$

を回転と平行移動によって標準形，

$$\mathrm{A}x^2+\mathrm{B}y^2=1$$

または，

$$y^2=4px$$

の形に帰着させることができます。

これを一般的に扱って作ったプログラムがリスト2です。係数を入力すると目的の曲線のグラフを標準形のものと同時に描きます。方程式によっては因数分解ができて直線が現れることもあります。

このようにして一般の二次曲線を描くことができますので，リスト1の一次変換による図形の移動にも，

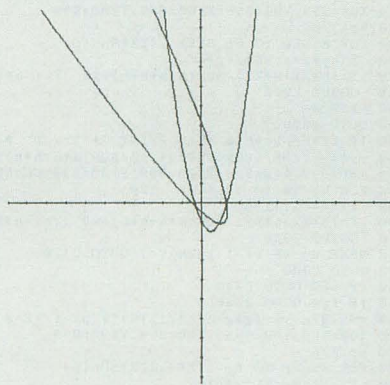
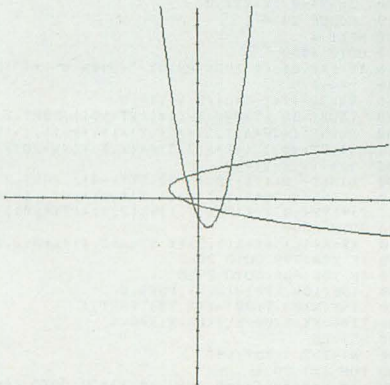
4)　一般の二次曲線

を付け加えることにしました。

さて，次回は不等式の表す領域を描いてみたいと思います。

リスト1　一次変換による曲線の移動

```
1000   INIT:WIDTH 80
1010   PRINT "     *******************************************************"
1020   PRINT "     *     ｲﾁｼﾞﾍﾝｶﾝ ﾆ ﾖﾙ ｷｮｸｾﾝ ﾉ ｲﾄﾞｳ                          *"
1030   PRINT "     *                                                     *"
1040   PRINT "     *                        X1/turbo  [ Hu-Basic ]       *"
1050   PRINT "     *                       1985/8/20    T.Yaso           *"
1060   PRINT "     *******************************************************"
1070   PAUSE 15:CLS
1080 PRINT "ｷｮｸｾﾝ ﾉ ﾎｳﾃｲｼｷ ｦ ｴﾗﾍﾞ｡"
1090 INPUT "1. ﾖｳｶﾝｽｳ y=f(x)  2. ｷｮｸﾎｳﾃｲｼｷ r=f(s)  3. ﾊﾞｲｶｲﾍﾝｽｳ x=f(t):y=g(t)  4
. ﾆｼﾞｷｮｸｾﾝ",Z :CLS
1100 ON Z GOTO 1110,1160,1210,1270
1110 PRINT "y=f(x) ｦ ﾃｲｷﾞ ｾﾖ｡"
1120 LIST 1140
1130 LOCATE 0,2:PRINT "GOTO 1140" :STOP
1140 DEF FNY(x)=2*X-3
1150 Z=1: GOTO 1320
1160 PRINT "r=f(s) ｦ ﾃｲｷﾞ ｾﾖ｡"
1170 LIST 1190
1180 LOCATE 0,2:PRINT "GOTO 1190" :STOP
1190 DEF FNr(s)=1
1200 Z=2:GOTO 1320
1210 PRINT "x=f(t):y=g(t) ｦ ﾃｲｷﾞ ｾﾖ｡"
1220 LIST 1240
1230 LOCATE 0,2:PRINT "GOTO 1240" :STOP
1240 DEF FNf(t)=1*(T-SIN(T))                    :DEF FNg(t)=1*(1-COS(t))
1250 INPUT "Tmin,Tmax";Tmin,Tmax
1260 Z=3:GOTO 1320
1270  PRINT "         2ｼﾞｷｮｸｾﾝ ﾉ ﾎｳﾃｲｼｷ   a・X^2+H・XY+b・Y^2+G・X+F・Y+c=0   ﾉ ｹｲｽｳ "
1280  INPUT "              a,H,b,G,F,c   ｦ INPUT ｾﾖ ";a,H,b,G,F,c
1290  A$(1)=STR$(a):A$(2)=STR$(H):A$(3)=STR$(b):A$(4)=STR$(G):A$(5)=STR$(F):
      A$(6)=STR$(c)
1300  h=H/2:g=G/2:f=F/2 :Z=4
1310 '
1320  CLS4:ON  ERROR GOTO 3310
1330  PRINT "ｷﾞｮｳﾚﾂ ﾉ ｾｲﾌﾞﾝ ｦ INPUT ｾﾖ｡"
1340  LOCATE 2,2 :INPUT "(1,1)=";M$(1,1):A(1,1)=CALC(M$(1,1))
1350  LOCATE 22,2:INPUT "(1,2)=";M$(1,2):A(1,2)=CALC(M$(1,2))
1360  LOCATE 2,4 :INPUT "(2,1)=";M$(2,1):A(2,1)=CALC(M$(2,1))
1370  LOCATE 22,4:INPUT "(2,2)=";M$(2,2):A(2,2)=CALC(M$(2,2))
1380  D=A(1,1)*A(2,2)-A(1,2)*A(2,1)
1390  IF D=0 THEN PRINT "ｷﾞｬｸｷﾞｮｳﾚﾂ ﾊ ｿﾝｻﾞｲ ｼﾅｲ"
1400  INIT:WIDTH 80:CLS4
1410 INPUT " ﾊﾝｲ ( Xn<X<Xm ):Xn,Xm";Xn,Xm
1420 INPUT "     ( Yn<Y<Ym ):Yn,Ym";Yn,Ym
1430 WINDOW (240,0)-(639,199),(Xn,Ym)-(Xm,Yn)
1440  LINE(Xn,0)-(Xm,0),PSET:LINE(0,Yn)-(0,Ym),PSET:Sx=(Xm-Xn)/20:Sy=(Ym-Yn)/20
1450  FOR I=0 TO 20:LINE(Xn+Sx*I,-.06*Sy)-(Xn+Sx*I,.06*Sy) ,PSET:
                    LINE(-.05*Sx,Yn+Sy*I)-(.05*Sx,Yn+Sy*I),PSET:
           NEXT I
1460 ON Z GOTO 1470,1520,1570,1620
1470 X00=999:Y00=999:XX00=999:YY00=999
1480 FOR X=-10 TO 10 STEP .125
1490 y=FNY(x) :GOSUB 2930
1500 NEXT x
1510 GOTO 3200
1520 X00=999:Y00=999:XX00=999:YY00=999
1530 FOR s=0 TO 2*π+.1  STEP .125
1540  x=FNr(s)*COS(s):y=FNr(s)*SIN(s):GOSUB 2930
1550 NEXT s
1560 GOTO 3200
1570 X00=999:Y00=999:XX00=999:YY00=999
1580 FOR t=Tmin TO Tmax STEP .125
1590  x=FNf(t):y=FNg(t):GOSUB 2930
1600 NEXT t
1610 GOTO 3200
1620 DTAA=a*b*c+2*h*f*g-g*g*b-a*f*f-c*h*h:DTA=a*b-h*h
1630 IF DTA=0 GOTO 2320
1640 x0=(f*h-b*g)/DTA:y0=(g*h-a*f)/DTA
1650 fx0y0=DTAA/DTA
1660 IF fx0y0=0 GOTO 2010
1670 IF a-b=0 THEN TT=π/2:t=π/4 ELSE TT=ATN(2*h/(a-b)):t=TT/2
1680 aa=((a+b)+(a-b)*COS(TT))/2+h*SIN(TT):bb=((a+b)-(a-b)*COS(TT))/2-h*SIN(TT)
1690 AAA=-aa/fx0y0:BBB=-bb/fx0y0
1700 IF AAA<0 AND BBB<0 THEN PRINT "ｷｮ ﾉ ﾀﾞｴﾝ":GOTO 3000
1710 IF AAA*BBB<0 GOTO 1810
1720 'ﾀﾞｴﾝ
1730 X00=999:Y00=999:XX00=999:YY00=999
1740 FOR s=0 TO 2*π +.1 STEP .125
1750  XX=SQR(1/AAA)*COS(s):YY=SQR(1/BBB)*SIN(s)
1760  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1770  GOSUB 2930
1780 NEXT s
1790 GOTO 3000
1800 'ｿｳｷｮｸｾﾝ
1810 IF BBB<0 GOTO 1910
1820 V=-1
1830 X00=999:Y00=999:XX00=999:YY00=999
1840 FOR s=-4 TO 4 STEP .125
1850  XX=SQR(1/ABS(AAA))*(EXP(s)-EXP(-s))/2:YY=V*SQR(1/ABS(BBB))*(EXP(s)+EXP(-s)
)/2
1860  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1870  GOSUB 2930
1880 NEXT s
1890 IF V=-1 THEN V=1:GOTO 1830
1900 GOTO 3000
1910 V=-1
1920 X00=999:Y00=999:XX00=999:YY00=999
1930 FOR s=-4 TO 4 STEP .125
1940  XX=V*SQR(1/ABS(AAA))*(EXP(s)+EXP(-s))/2:YY=SQR(1/ABS(BBB))*(EXP(s)-EXP(-s)
)/2
1950  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1960  GOSUB 2930
1970 NEXT s
1980 IF V=-1 THEN V=1:GOTO 1920
1990 GOTO 3000
2000 'ﾁｮｸｾﾝ
2010 IF DTA>0 THEN PRINT "ｷｮ ﾉ ﾆﾁｮｸｾﾝ":GOTO 3000
2020 IF b=0 GOTO 2130
```

グラフ1

グラフ2

```
2030 L1=(-h+SGN(-h+.000001)*SQR(-DTA))/b :L2=a/(b*L1):GOTO 2040
2040 V=-1:L=L1
2050 X00=999:Y00=999:XX00=999:YY00=999
2060 FOR s=-10 TO 10 STEP .125
2070 XX=s :YY=L*s
2080  X=XX+x0:Y=YY+y0
2090  GOSUB 2930
2100  NEXT s
2110 IF V=-1 THEN V=1:L=L2:GOTO 2050
2120 GOTO 3000
2130 IF A=0 GOTO 2250 ELSE LINE(x0,0)-(x0,399),PSET,2
2140   LINE((A(1,1)*x0-10*A(1,2)),(A(2,1)*x0-10*A(2,2)))-((A(1,1)*x0+10*A(1,2)),
(A(2,1)*x0+10*A(2,2))),PSET,4
2150 IF h=0 GOTO 2230
2160 X00=999:Y00=999:XX00=999:YY00=999
2170 FOR s=-10 TO 10 STEP .125
2180  XX=s:YY=-a/(2*h)*s
2190  X=XX+x0:Y=YY+y0
2200  GOSUB 2930
2210 NEXT s
2220 GOTO 3000
2230  PRINT "ｲｯﾁｮｸｾﾝ"
2240 GOTO 3000
2250 IF H=0 THEN PRINT "ｾﾞﾝﾍｲﾒﾝ":GOTO 3000
2260  LINE(x0,-10)-(x0,10),PSET,2
2270   LINE((A(1,1)*x0-10*A(1,2)),(A(2,1)*x0-10*A(2,2)))-((A(1,1)*x0+10*A(1,2)),
(A(2,1)*x0+10*A(2,2))),PSET,4
2280  LINE(-10,y0)-(10,y0),PSET,2
2290   LINE((-10*A(1,1)+A(1,2)*y0),(-10*A(2,1)+A(2,2)*y0))-((10*A(1,1)+A(1,2)*y0
),(10*A(2,1)+A(2,2)*y0)),PSET,4
2300 GOTO 3000
2310 'ﾎｳﾌﾞﾂｾﾝ
2320 IF h=0 GOTO 2540
2330 bb=(a*a*a+2*a*h*h+b*h*h)/(a*a+h*h):gg=(g*h-a*f)/SQR(a*a+h*h):
     ff=(a*g+f*h)/SQR(a*a+h*h)
2340 IF gg=0 GOTO 2450
2350 yy0=-ff/bb:xx0=ff*ff/(2*gg*bb)-c/(2*gg):p=-gg/(2*bb)
2360 x0=(h*xx0+a*yy0)/SQR(a*a+h*h):y0=(-a*xx0+h*yy0)/SQR(a*a+h*h)
2370 X00=999:Y00=999:XX00=999:YY00=999
2380 Ep=5/p
2390 FOR s=-Ep TO Ep STEP .125*SGN(p)
2400  XX=p*s*s:YY=2*p*s
2410  X=(h*XX+a*YY)/SQR(a*a+h*h)+x0 :Y=(-a*XX+h*YY)/SQR(a*a+h*h)+y0
2420  GOSUB 2930
2430 NEXT s
2440 GOTO 3000
2450 IF ff*ff-bb*c<0 THEN PRINT "ｼﾞﾂﾃﾝ ｶﾞ ﾅｲ":GOTO 3000
2460 yy0=-ff/bb:xx0=0:x0=a*yy0/SQR(a*a+h*h):y0=h*yy0/SQR(a*a+h*h):V=-1
2470 X00=999:Y00=999:XX00=999:YY00=999:CC=SQR(ff*ff-bb*c)/bb
2480 FOR s=-10 TO 10 STEP .125
2490  YY=V*CC:XX=s
2500  X=(h*XX+a*YY)/SQR(a*a+h*h)+x0 :Y=(-a*XX+h*YY)/SQR(a*a+h*h)+y0
2510  GOSUB 2930
2520 NEXT s :IF V=-1 THEN V=1:GOTO 2470
2530 GOTO 3000
2540 IF a=0 GOTO 2740
2550 IF f=0 GOTO 2650
2560 x0=-g/a:y0=(g*g-a*c)/(2*a*f):p=-f/(2*a)
2570 X00=999:Y00=999:XX00=999:YY00=999
2580 Ep=5/p
2590 FOR s=-Ep TO Ep STEP .125*SGN(p)
2600  YY=p*s*s:XX=2*p*s
2610  X=XX+x0 :Y=YY+y0
2620  GOSUB 2930
2630 NEXT s
2640 GOTO 3000
2650 IF g*g-a*c<0 THEN PRINT "ｼﾞﾂﾃﾝ ｶﾞ ﾅｲ":GOTO 3000
2660  x0=-g/a
2670  XX=SQR(g*g-a*c)/a:X=XX+x0
2680  LINE((XX+x0),-10)-((XX+x0),10),PSET,2
2690   LINE((A(1,1)*(XX+x0)-10*A(1,2)),(A(2,1)*(XX+x0)-10*A(2,2)))-((A(1,1)*(XX+
x0)+10*A(1,2)),(A(2,1)*(XX+x0)+10*A(2,2))),PSET,4
2700  XX=-XX:X=XX+x0
2710  LINE((XX+x0),-10)-((XX+x0),10),PSET,2
2720   LINE((A(1,1)*(XX+x0)-10*A(1,2)),(A(2,1)*(XX+x0)-10*A(2,2)))-((A(1,1)*(XX+
x0)+10*A(1,2)),(A(2,1)*(XX+x0)+10*A(2,2))),PSET,4
2730 GOTO 3000
2740 IF g=0 GOTO 2840
2750 y0=-f/b:x0=(f*f-b*c)/(2*b*g):p=-g/(2*b)
2760 X00=999:Y00=999:XX00=999:YY00=999
2770 Ep=5/p
2780 FOR s=-Ep TO Ep STEP .125*SGN(p)
2790  XX=p*s*s:YY=2*p*s
2800  X=XX+x0 :Y=YY+y0
2810  GOSUB 2930
2820 NEXT s
2830 GOTO 3000
2840 IF f*f-b*c<0 THEN PRINT "ｼﾞﾂﾃﾝ ｶﾞ ﾅｲ":GOTO 3000
2850  y0=-f/b
2860  YY=SQR(f*f-b*c)/b:Y=YY+y0
2870  LINE(-10,(YY+y0))-(10,(YY+y0)),PSET,2
2880   LINE((-10*A(1,1)+A(1,2)*(YY+y0)),(-10*A(2,1)+A(2,2)*(YY+y0)))-((10*A(1,1)
+A(1,2)*(YY+y0)),(10*A(2,1)+A(2,2)*(YY+y0))),PSET,4
2890  YY=-YY:Y=YY+y0
2900  LINE(-10,(YY+y0))-(10,(YY+y0)),PSET,2
2910   LINE((-10*A(1,1)+A(1,2)*(YY+y0)),(-10*A(2,1)+A(2,2)*(YY+y0)))-((10*A(1,1)
+A(1,2)*(YY+y0)),(10*A(2,1)+A(2,2)*(YY+y0))),PSET,4
2920 GOTO 3000
2930  XX=A(1,1)*X+A(1,2)*Y:YY=A(2,1)*X+A(2,2)*Y
2940 IF X00=999 GOTO 2980
2950 IF Y00=999 GOTO 2980
2960 LINE(X00,Y00)-(X,Y),PSET,2
2970 LINE(XX00,YY00)-(XX,YY),PSET,4
2980 XX00=XX:YY00=YY:X00=X:Y00=Y
2990 RETURN
3000 PRINT "    ﾎｳﾃｲｼｷ"
3010 FOR I=1 TO 6
3020  IF VAL(A$(I))=0 THEN A$(I)="":GOTO 3060
3030  IF LEFT$(A$(I),1)=" " THEN A$(I)="+"+RIGHT$(A$(I),LEN(A$(I))-1)
3040  IF VAL(A$(I))=1 THEN A$(I)="+"
3050  IF VAL(A$(I))=-1 THEN A$(I)="-"
```

グラフ3

グラフ4

$x^2+2xy+3y^2-x-2y-6=0$

逆行列が存在しないので楕円上の点はすべてひとつの線分に移る

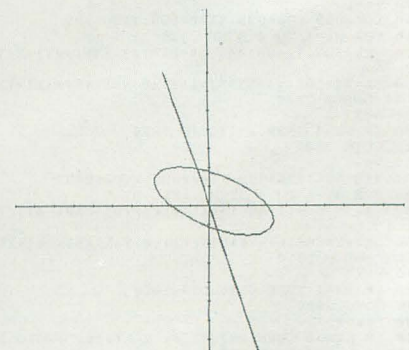
範囲（$Xn<X<Xm$）：Xn, Xm？ −10, 10

（$Yn<Y<Ym$）：Yn, Ym？ −10, 10

中心の座標（.25, .25）

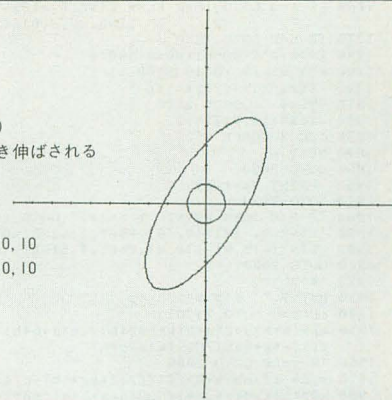
行　列

$$\begin{pmatrix} 1 & 2 \\ -3 & -6 \end{pmatrix}$$

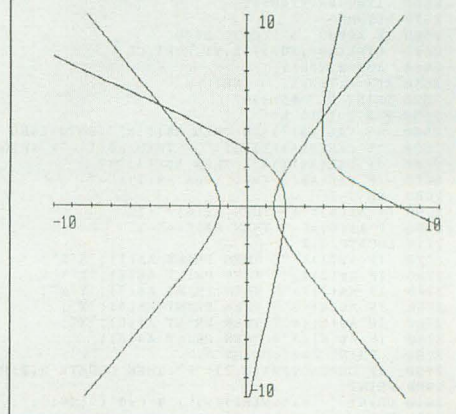
▶「夕やけニャンニャン」が８月で終わるそうである。我が愛機Ｘ１turboZのビデオデジタイズ機能はすることがなくなってしまうではないか。　佐藤　健史（22）東京都

```
3060  NEXT I
3070 IF A$(6)="+" THEN A$(6)="+1"
3080 IF A$(6)="-" THEN A$(6)="-1"
3090 LOCATE 3,3
3100  IF A$(1)<>"" THEN PRINT A$(1);"X^2";
3110  IF A$(2)<>"" THEN PRINT A$(2);"X·Y";
3120  IF A$(3)<>"" THEN PRINT A$(3);"Y^2";
3130  IF A$(4)<>"" THEN PRINT A$(4);"X";
3140  IF A$(5)<>"" THEN PRINT A$(5);"Y";
3150  IF A$(6)<>"" THEN PRINT A$(6);
3160   PRINT "=0":PRINT
3170  IF CHARACTER$(3,3)="+" THEN LOCATE 3,3:PRINT " "
3180 PRINT
3190 PRINT "  ﾁｭｳｼﾝ ﾉ ｻﾞﾋｮｳ (";x0;",";y0;")"
3200 PRINT "ｷﾞｮｳﾚﾂ"
3210 L=1
3220 FOR I=1 TO 2:FOR J=1 TO 2
3230  IF L<LEN(M$(I,J)) THEN L=LEN(M$(I,J))
3240 NEXT:NEXT
3250 PRINT "┌";TAB(2*L+4);"┐"
3260 PRINT "| ";M$(1,1);TAB(L+3);M$(1,2);TAB(2*L+4);"|"
3270 PRINT "|";TAB(2*L+4);"|"
3280 PRINT "| ";M$(2,1);TAB(L+3);M$(2,2);TAB(2*L+4);"|"
3290 PRINT "└";TAB(2*L+4);"┘"
3300 END
3310 Y00=999:Y=999
3320 RESUME NEXT
```

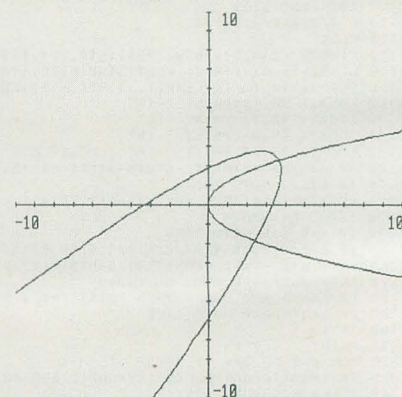
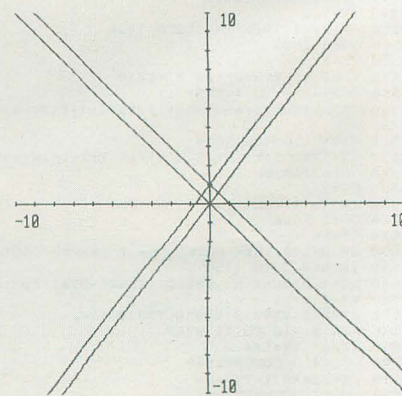
グラフ5

$r=1$
原点を中心とする半径1の円
固有値 $\lambda_1=5,\ \lambda_1=-2$
固有ベクトル $\vec{v_1}=(3,4),\ \vec{v_2}=(1,-1)$
$\vec{v_1}$の方向に5倍，$\vec{v_2}$の方向に2倍に引き伸ばされる

範囲 $(Xn<X<Xm)$：Xn, Xm？ $-10, 10$
　　 $(Yn<Y<Ym)$：Yn, Ym？ $-10, 10$
行 列
$\begin{bmatrix} 1 & 3 \\ 4 & 2 \end{bmatrix}$

リスト2 二次曲線のグラフ(標準形への変換)

```
1000 INIT:WIDTH 80
1010 PRINT "    ********************************************************"
1020 PRINT "    *     2ｼﾞｷｮｸｾﾝ ﾉ ｸﾞﾗﾌ  (ﾋｮｳｼﾞｭﾝｹｲ ﾍﾉ ﾍﾝｶﾝ)              *"
1030 PRINT "    *                                                      *"
1040 PRINT "    *                       X1/turbo  [ Hu-Basic ]         *"
1050 PRINT "    *                      1987/3/20    T.Yaso             *"
1060 PRINT "    ********************************************************"
1070 PAUSE 15:CLS
1080 PRINT "          2ｼﾞｷｮｸｾﾝ ﾉ ﾎｳﾃｲｼｷ     a·X^2+H·XY+b·Y^2+G·X+F·Y+c=0    ﾉ ｹｲｽｳ "
1090 INPUT "              a,H,b,G,F,c    ｦ INPUT ｾﾖ ";a,H,b,G,F,c
1100 A$(1)=STR$(a):A$(2)=STR$(H):A$(3)=STR$(b):A$(4)=STR$(G):A$(5)=STR$(F):
     A$(6)=STR$(c)
1110 h=H/2:g=G/2:f=F/2
1120 CLS4
1130 WINDOW (240,0)-(639,199),(-10,10)-(10,-10)
1140 LINE(-10,0)-(10,0),PSET:LINE(0,10)-(0,-10),PSET
1150 FOR I=-10 TO 10:LINE(I,.1)-(I,-.1),PSET:LINE(-.1,I)-(.1,I),PSET:NEXT I
1160 LOCATE 30,13:PRINT "-10"
1170 LOCATE 78,13:PRINT "10"
1180 LOCATE 55,0 :PRINT " 10"
1190 LOCATE 55,24:PRINT "-10";:LOCATE 0,0
1200 DTAA=a*b*c+2*h*f*g-g*g*b-a*f*f-c*h*h:DTA=a*b-h*h
1210 IF DTA=0 GOTO 1930
1220 x0=(f*h-b*g)/DTA:y0=(g*h-a*f)/DTA
1230 fx0y0=DTAA/DTA
1240 IF fx0y0=0 GOTO 1630
1250 IF a-b=0 THEN TT=π/2:t=π/4 ELSE TT=ATN(2*h/(a-b)):t=TT/2
1260 aa=((a+b)+(a-b)*COS(TT))/2+h*SIN(TT):bb=((a+b)-(a-b)*COS(TT))/2-h*SIN(TT)
1270 AAA=-aa/fx0y0:BBB=-bb/fx0y0
1280 IF AAA<0 AND BBB<0 THEN PRINT "ｷｮ ﾉ ﾀﾞｴﾝ":GOTO 2620
1290 IF AAA*BBB<0 GOTO 1400
1300 'ﾀﾞｴﾝ
1310 PRINT "   ﾀﾞｴﾝ"
1320 FOR s=0 TO 2*π +.1 STEP .1
1330  XX=SQR(1/AAA)*COS(s):YY=SQR(1/BBB)*SIN(s)
1340  CL=4  :GOSUB 2540
1350  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1360  CL=2:GOSUB 2580
1370 NEXT s
1380 GOTO 2620
1390 'ｿｳｷｮｸｾﾝ
1400 PRINT " ｿｳｷｮｸｾﾝ"
1410 IF BBB<0 GOTO 1520
1420 V=-1
1430 X00=0:Y00=0:XX00=0:YY00=0
1440 FOR s=-4 TO 4 STEP .1
1450  XX=SQR(1/ABS(AAA))*(EXP(s)-EXP(-s))/2:YY=V*SQR(1/ABS(BBB))*(EXP(s)+EXP(-s)
)/2
1460  CL=4  :GOSUB 2540
1470  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1480  CL=2:GOSUB 2580
1490 NEXT s
1500 IF V=-1 THEN V=1:GOTO 1430
1510 GOTO 2620
1520 V=-1
1530 XX00=0:YY00=0:X00=0:Y00=0
1540 FOR s=-4 TO 4 STEP .1
1550  XX=V*SQR(1/ABS(AAA))*(EXP(s)+EXP(-s))/2:YY=SQR(1/ABS(BBB))*(EXP(s)-EXP(-s)
)/2
1560 CL=4  :GOSUB 2540
1570  X=XX*COS(t)-YY*SIN(t)+x0:Y=XX*SIN(t)+YY*COS(t)+y0
1580  CL=2:GOSUB 2580
1590 NEXT s
1600 IF V=-1 THEN V=1:GOTO 1530
1610 GOTO 2620
1620 'ﾁｮｸｾﾝ
1630 IF DTA>0 THEN PRINT "ｷｮ ﾉ ﾆﾁｮｸｾﾝ":GOTO 2620
1640 IF b=0 GOTO 1760
1650 L1=(-h+SGN(-h+.000001)*SQR(-DTA))/b :L2=a/(b*L1):GOTO 1660
1660 V=-1:L=L1
1670 XX00=0:YY00=0:X00=0:Y00=0
1680 FOR s=-10 TO 10 STEP .1
1690 XX=s :YY=L*s
1700  CL=4  :GOSUB 2540
1710  X=XX+x0:Y=YY+y0
1720  CL=2:GOSUB 2580
1730  NEXT s
```

グラフ6

```
1740 IF V=-1 THEN V=1:L=L2:GOTO 1670
1750 GOTO 2620
1760 IF A=0 GOTO 1880 ELSE LINE(0,-10)-(0,10 ),PSET,4:
                            LINE(x0,-10)-(x0,10),PSET,2
1770 IF h=0 GOTO 1860
1780 XX00=0:YY00=0:X00=0:Y00=0
1790 FOR s=-10 TO 10 STEP .1
1800  XX=s:YY=-a/(2*h)*s
1810  CL=4  :GOSUB 2540
1820  X=XX+x0:Y=YY+y0
1830  CL=2:GOSUB 2580
1840 NEXT s
1850 GOTO 2620
1860  PRINT "ｲｯﾁｮｸｾﾝ"
1870 GOTO 2620
1880 IF H=0 THEN PRINT "ｾﾞﾝﾍｲﾒﾝ":GOTO 2620
1890  LINE(0,-10)-(0,10),PSET,4:LINE(x0,-10)-(x0,10),PSET,2
1900  LINE(-10,0)-(10,0),PSET,4:LINE(-10,y0)-(10,y0),PSET,2
1910 GOTO 2620
1920 'ﾎｳﾌﾞﾂｾﾝ
1930 PRINT "  ﾎｳﾌﾞﾂｾﾝ"
1940 IF h=0 GOTO 2170
1950 bb=(a*a*a+2*a*h*h+b*h*h)/(a*a+h*h):gg=(g*h-a*f)/SQR(a*a+h*h):
     ff=(a*g+f*h)/SQR(a*a+h*h)
1960 IF gg=0 GOTO 2080
1970 yy0=-ff/bb:xx0=ff*ff/(2*gg*bb)-c/(2*gg):p=-gg/(2*bb)
1980 x0=(h*xx0+a*yy0)/SQR(a*a+h*h):y0=(-a*xx0+h*yy0)/SQR(a*a+h*h)
1990 XX00=0:YY00=0:X00=0:Y00=0
2000 Ep=5/p
2010 FOR s=-Ep TO Ep STEP .1*SGN(p)
2020  XX=p*s*s:YY=2*p*s
2030 CL=4:GOSUB 2540
2040  X=(h*XX+a*YY)/SQR(a*a+h*h)+x0:Y=(-a*XX+h*YY)/SQR(a*a+h*h)+y0
2050 CL=2:GOSUB 2580
2060 NEXT s
2070 GOTO 2620
2080 IF ff*ff-bb*c<0 THEN PRINT "ｼﾞｯﾃﾝ ｶﾞ ﾅｲ":GOTO 2620
2090 yy0=-ff/bb:xx0=0:x0=a*yy0/SQR(a*a+h*h):y0=h*yy0/SQR(a*a+h*h):V=-1
2100 XX00=0:YY00=0:X00=0:Y00=0 :CC=SQR(ff*ff-bb*c)/bb
2110 FOR s=-10 TO 10 STEP .1
2120  YY=V*CC:XX=s:CL=4:GOSUB 2540
2130  X=(h*XX+a*YY)/SQR(a*a+h*h)+x0:Y=(-a*XX+h*YY)/SQR(a*a+h*h)+y0
2140 CL=2:GOSUB 2580
2150 NEXT s :IF V=-1 THEN V=1:GOTO 2100
2160 GOTO 2620
2170 IF a=0 GOTO 2360
2180 IF f=0 GOTO 2290
2190 x0=-g/a:y0=(g*g-a*c)/(2*a*f):p=-f/(2*a)
2200 XX0=0:YY00=0:X00=0:Y00=0
2210 Ep=5/p
2220 FOR s=-Ep TO Ep STEP .1*SGN(p)
2230  YY=p*s*s:XX=2*p*s
2240 CL=4:GOSUB 2540
2250  X=XX+x0:Y=YY+y0
2260 CL=2:GOSUB 2580
2270 NEXT s
2280 GOTO 2620
2290 IF g*g-a*c<0 THEN PRINT "ｼﾞｯﾃﾝ ｶﾞ ﾅｲ":GOTO 2620
2300  x0=-g/a
2310  XX=SQR(g*g-a*c)/a:X=XX+x0
2320  LINE (XX,-10)-(XX,10),PSET,4:LINE(XX+x0,-10)-(XX+x0,10),PSET,2
2330  XX=-XX:X=XX+x0
2340  LINE (XX,-10)-(XX,10),PSET,4:LINE(XX+x0,-10)-(XX+x0,10),PSET,2
2350 GOTO 2620
2360 IF g=0 GOTO 2470
2370 y0=-f/b:x0=(f*f-b*c)/(2*b*g):p=-g/(2*b)
2380 XX00=0:YY00=0:X00=0:Y00=0
2390 Ep=5/p
2400 FOR s=-Ep TO Ep STEP .1*SGN(p)
2410  XX=p*s*s:YY=2*p*s
2420 CL=4:GOSUB 2540
2430  X=XX+x0:Y=YY+y0
2440 CL=2:GOSUB 2580
2450 NEXT s
2460 GOTO 2620
2470 IF f*f-b*c<0 THEN PRINT "ｼﾞｯﾃﾝ ｶﾞ ﾅｲ":GOTO 2620
2480  y0=-f/b
2490  YY=SQR(f*f-b*c)/b:Y=YY+y0
2500  LINE (-10,YY)-(10,YY),PSET,4:LINE(-10,YY+y0)-(10,YY+y0),PSET,2
2510  YY=-YY:Y=YY+y0
2520  LINE (-10,YY)-(10,YY),PSET,4:LINE(-10,YY+y0)-(10,YY+y0),PSET,2
2530 GOTO 2620
2540 IF XX00=0     GOTO 2560
2550  LINE(XX00,YY00)-(XX,YY),PSET,CL
2560  XX00=XX:YY00=YY
2570 RETURN
2580 IF X00=0      GOTO 2600
2590  LINE(X00,Y00)-(X,Y),PSET,CL
2600  X00=X:Y00=Y
2610 X00=X:Y00=Y:    RETURN
2620 PRINT "   ﾎｳﾃｲｼｷ"
2630 FOR I=1 TO 6
2640  IF VAL(A$(I))=0 THEN A$(I)="":GOTO 2680
2650  IF LEFT$(A$(I),1)=" " THEN A$(I)="+"+RIGHT$(A$(I),LEN(A$(I))-1)
2660  IF VAL(A$(I))=1 THEN A$(I)="+"
2670  IF VAL(A$(I))=-1 THEN A$(I)="-"
2680  NEXT I
2690 IF A$(6)="+" THEN A$(6)="+1"
2700 IF A$(6)="-" THEN A$(6)="-1"
2710 LOCATE 3,2
2720  IF A$(1)<>"" THEN PRINT A$(1);"X^2";
2730  IF A$(2)<>"" THEN PRINT A$(2);"X·Y";
2740  IF A$(3)<>"" THEN PRINT A$(3);"Y^2";
2750  IF A$(4)<>"" THEN PRINT A$(4);"X";
2760  IF A$(5)<>"" THEN PRINT A$(5);"Y";
2770  IF A$(6)<>"" THEN PRINT A$(6);
2780   PRINT "=0":PRINT
2790  IF CHARACTER$(3,2)="+" THEN LOCATE 3,2:PRINT " "
2800 PRINT
2810 PRINT "  ﾁｭｳｼﾝ(ﾁｮｳﾃﾝ) ﾉ ｻﾞﾋｮｳ (";x0;",";y0;")"
2820 END
```

グラフ7

グラフ8

THE SENTINEL

今月はパズルゲーム「碁石拾い」とついに現れたS-OS上で漢字を出力する試み「JACKWRITE」，そして予告どおりFM-7/77版“SWORD”の豪華3本立てでお送りします。

碁石拾いは小さいながら非常によくまとまったパズルゲームです。もちろんFM版でも大丈夫。手軽に打ち込める大きさですから“SWORD”の動作確認にも使えますね。このところアドベンチャーやリアルタイムゲームが続きましたが，久々に頭の体操を楽しむのもよいのではないでしょうか。

●漢字出力パッケージ

なんと漢字ROMの代わりにディスクドライブを使い，ディスク上にある漢字データをMAGICを使って画面に書き出すという荒技による漢字出力パッケージです。フォントデータではなくベクトルによるストロークデータを使用することによりMAGICのラインコマンドで高速な表示を行います。MAGICの面白い使用例だと思いませんか。これを応用すれば漢字だけでなくゲームのキャラクターを描いたり，さまざまな書体の文字を表示したりと，いろいろ遊べそうですね。漢字の使い道はなにもワープロだけとは限りません。もっと楽しく，新しい視野に立ったものもあっていいはずです。

そのうえ，このパッケージはディスクドライブとMAGICを使用することにより，現在のS-OSとMAGICの仕様のままでも比較的簡単に全機種で共通のイメージの画面出力を得ることができます。加えてごく一般に行われているような漢字フォントを用いる方法では実現しにくい自由なレイアウトができるなどの長所も持っているのです。まだシステムとして完備された環境にはありませんが，大いなる可能性を秘めた試みといえるでしょう。

●FM-7/77版S-OS“SWORD”

ついにFM-7/77シリーズ版のS-OS“SWORD”を発表することができました。今回発表するのは基本的にZ80カード版ですが，Z80カードを持っていない方もエミュレータを通すことにより，カード版に若干の修正点を加えるだけでそのまま使用することができます。Z80カード版を作ったのはFMユーザー，エミュレータを作ったのはS1ユーザーとなかなかユニークな取り合わせですが，さまざまな機種のユーザーが協力してひとつのシステムを作るというのもS-OSならではかもしれませんね。

エミュレータというのはZ80のインタプリタのようなものですから速度的に苦しい面もあります(4MHz Z80マシンの約1/40)。今回のエミュレータ版はひとつの例にすぎません。エミュレータを改善する，“SWORD”本体を高速化する，マシン語のコンバータを作るなど，あとは実際にこの“SWORD”を使用する皆さん自身が工夫してみてください。Z80カード版も6809システムに依存しないようにすればかなり速くなるはずです。

さて，今月のFM-7/77版に続き来月はPC-8001mkII/8801用S-OS“SWORD”オールRAMバージョンの発表を予定しています。PC-8001mkII版は64Kバイトに拡張したPC-8001でも使用できます。PC-8801版は高速化しての再登場です。PCユーザーの方お楽しみに。

第47部　パズルゲーム碁石拾い
第48部　漢字出力パッケージJACKWRITE
特別付録　FM-7/77版S-OS“SWORD”

全機種共通システム掲載記事

■85年6月号
序論　共通化の試み
第1部　S-OS“MACE”
第2部　Lisp-85インタプリタ
第3部　チェックサムプログラム
■85年7月号
第4部　マシン語プログラム開発入門
第5部　エディタアセンブラZEDA
第6部　デバッグツールZAID
■85年8月号
第7部　ゲーム開発パッケージBEMS
第8部　ソースジェネレータZING
■85年9月号
インタラプト　S-OS番外地
第9部　マシン語入力ツールMACINTO-S
第10部　Lisp-85入門(1)
■85年10月号
第11部　仮想マシンCAP-X85
連載　Lisp-85入門(2)
■85年11月号
連載　Lisp-85入門(3)
■85年12月号
第12部　Prolog-85発表
■86年1月号
第13部　リロケータブルのお話
第14部　FM音源サウンドエディタ
■86年2月号
第15部　S-OS“SWORD”
第16部　Prolog-85入門(1)
■86年3月号
第17部　magiFORTH発表
連載　Prolog-85入門(2)
■86年4月号
第18部　思考ゲームJEWEL
第19部　LIFE GAME
連載　基礎からのmagiFORTH
連載　Prolog-85入門(3)
■86年5月号
第20部　スクリーンエディタE-MATE
連載　実戦演習magiFORTH
■86年6月号
第21部　Z80TRACER
第22部　magiFORTH TRACER
第23部　ディスクダンプ&エディタ
第24部　“SWORD”2000 QD
連載　対話で学ぶmagiFORTH
特別付録　PC-8801版S-OS“SWORD”
■86年7月号
第25部　FM音源ミュージックシステム
付録　FM音源ボードの製作
連載　計算力アップのmagiFORTH
特別付録　SMC-777版S-OS“SWORD”
■86年8月号
第26部　対局五目並べ
第27部　MZ-2500版S-OS“SWORD”
■86年9月号
第28部　FuzzyBASIC発表
連載　明日に向かってmagiFORTH
■86年10月号
第29部　ちょっと便利な拡張プログラム
第30部　ディスクモニタDREAM
第31部　FuzzyBASIC料理法<1>
■86年11月号
第32部　パズルゲームHOTTAN
第33部　MAZE in MAZE
連載　FuzzyBASIC料理法<2>
■86年12月号
第34部　CASL & COMET
連載　FuzzyBASIC料理法<3>
■87年1月号
第35部　マシン語入力ツールMACINTO-C
連載　FuzzyBASIC料理法<4>
■87年2月号
第36部　アドベンチャーゲームMARMALADE
第37部　テキアベ作成ツールCONTEX
■87年3月号
第38部　魔法使いはアニメがお好き
第39部　アニメーションツールMAGE
付録　“SWORD”再掲載とMAGICの標準化
■87年4月号
第40部　INVADER GAME
第41部　TANGERINE
■87年5月号
第42部　S-OS“SWORD”変身セット
第43部　MZ-700用“SWORD”をQD対応に
■87年6月号
インタラプト　コンパイラ物語
第44部　FuzzyBASICコンパイラ
第45部　エディタアセンブラZEDA-3
■87年7月号
第46部　STORY MASTER

＊以上のアプリケーションは，基本システムであるS-OS“MACE”またはS-OS“SWORD”がないと動作しませんのでご注意ください。

全機種共通S-OS"SWORD"要

パズルゲーム 碁石拾い

Hanai Akiyoshi
花井 章能

久々にS-OS用のパズルゲームの登場です。なかなかいうことを聞いてくれないテイカーを操り，画面にちりばめられた碁石を拾い集めてください。一見シンプルで簡単そうなゲームですが，意外にこれが手強いのです。面は20面用意されています。それでは全問征覇めざしてがんばってください。

パズルに挑戦

このプログラムはS-OS"SWORD"用のパズルゲームです。一般に「碁石拾い」というのは，盤上にある碁石を一定の規則に従ってすべて拾っていくというゲームなのですが，これをパソコン上に持ってきたのがこのプログラムです。しかし，この碁石拾いでは画面上にあるのは碁石ばかりではありません。知恵をしぼって難問にチャレンジしてみてください。

入力方法

リスト1のダンプリストを各機種のマシン語モニタまたはマシン語入力ツールから打ち込んでください。FM-7エミュレータ版の場合はシステムジェネレートの際にマシン語入力ツールを組み込みますので，"SWORD"上からJ3000でマシン語入力ツールを起動し，A000Hから入力してください。ダンプリストには縦サム横サムのほかに16進4桁のCRCチェックバイトがついていますが，FM-7ではとりあえずこの部分は無視しておいてもかまいません。

入力後は必ずセーブし，チェックサムとCRCチェックバイトをすべて確認してから起動するようにしてください。プログラム部分はかなり短いものですからソースリストで入力するのもよいかもしれません。

なお，実行先頭アドレスはA000Hです。あらかじめ画面を40字モードに設定しておき，"SWORD"上から JA000H で起動してください。

このプログラムはA000HからA31EHまでがプログラム，A31FH以降がデータ部です。データは画面をそのままの形で格納しているだけですから，データ構造はソースリスト（リスト3）の後半部を見ていただければだいたいわかると思います。自分で新しい面を作りたいというときには古いデータの上に新しいデータを読み込んでいけばよいでしょう。

ルールと遊び方

画面上に"<，>，A，V"という形で表示されているのがあなたの操るテイカーです。それぞれ先の尖っているほうが前になっています。このテイカーは次のようなキー操作により，

L	右方向に進む
I	上方向に進む
J	左方向に進む
K	下方向に進む

という動作をし，なにかにぶつかるまで止まらずに進みます。ぶつかった相手が碁石("＊")だった場合，その碁石は取られたことになりテイカーは碁石のあった位置で停止します。テイカーは後ろ方向には移動できません。進行方向に対して前進するか左右に動くかしかできないのです。したがって1手間違えるとそれだけでハマッてしまうこともあります。そういったときはRまたはBのキーを押してください。Rならその面の最初から，Bならひとつ前の面からやりなおすことができます。

このゲームの目的は画面上の碁石をすべて拾い尽くすことですが，画面には碁石のほかに，

■(壁)
　ぶつかると止まる。押しても動かない

@(石)
　ぶつかると止まる。押して動かすこともできる

O(穴)
　この穴に落ちるとその面の最初に戻ってしまう。石を落として埋めることができる

といったキャラクターが配置されています。これらの障害を乗り越えてみごと20面をクリアしてください。

Profile
◇花井君は福島県にお住まいの16歳，高校2年生です。マイコン歴は約4年，X1Cのユーザーです。現在，S-OS用の3Dシューティングゲームを制作中とか。

リスト1　碁石拾いダンプリスト

```
A000 3E 01 32 DA A0 3E 0C CD : 02
A008 F4 1F 11 DB A0 06 0B 1A : CA
A010 6F 13 1A 67 13 CD 1E 20 : 21
A018 CD E5 1F 21 09 00 19 54 : 68
A020 5D 10 EC 21 03 0A CD 1E : 72
A028 20 11 C7 A0 CD E5 1F 21 : 8A
A030 03 15 CD 1E 20 CD E5 1F : F4
A038 3A DA A0 3D 3C 20 05 3E : 90
A040 0C C3 F4 1F 06 00 0E 0A : 00

A048 16 30 FE 0A 38 04 04 91 : 1F
A050 18 F8 21 11 07 CD 1E 20 : 54
A058 82 CD F4 1F 3E 1D CD F4 : 7E
A060 1F CD F4 1F 78 82 CD F4 : BA
A068 1F AF 32 62 A2 3A DA A0 : B8
A070 47 21 7F A2 11 A0 00 19 : 53
A078 10 FD 44 4D 21 00 0B CD : 97
---------------------------------
SUM: 79 7A 8C 22 57 37 D3 20 D65F

A080 1E 20 26 0B 2E 04 CD E2 : 50
A088 1F 20 20 20 7B 00 0A 03 : 07
A090 CD F4 1F FE 2A 20 06 E5 : 13
A098 21 62 A2 34 E1 FE 3E 20 : 96
A0A0 08 7D 32 A8 A1 7C 32 A9 : 57
A0A8 A1 2C 7D FE 14 28 02 18 : 9E
A0B0 DD CD E2 1F 7B 0D 00 24 : 57
A0B8 7C FE 15 28 02 18 C5 3E : D4
A0C0 02 32 A7 A1 C3 53 A1 7B : AE
```

```
A0C8 7B 7B 7B 7B 7B 7B 7B 7B : D8
A0D0 7B 7B 7B 7B 7B 7B 7B 7B : D8
A0D8 7B 00 00 04 04 BA DE B2 : CD
A0E0 BC 20 CB DB B2 00 0A 07 : 45
A0E8 52 4F 55 4E 44 3A 20 20 : 02
A0F0 00 19 06 3E 20 54 41 4B : 5D
A0F8 45 52 20 00 19 08 2A 20 : 22
----------------------------------
SUM: F3 0C 90 4C D2 84 1E C2 4BA8

A100 47 4F 49 53 48 49 00 19 : DC
A108 0A 7B 20 57 41 4C 4C 20 : F5
A110 20 00 19 0C 40 20 53 54 : 4C
A118 4F 4E 45 20 00 19 0E 4F : 78
A120 20 48 4F 4C 45 20 20 00 : 88
A128 1A 11 49 1F 1F 1F 1F 1D : 0D
A130 4B 20 00 18 13 4A 20 3E : 3E
A138 20 4C 20 52 20 00 1F 13 : 30
A140 3A 52 45 50 4C 41 43 45 : 36
A148 00 1E 15 42 3A 42 41 43 : 75
A150 4B 00 00 CD D7 A1 E5 32 : A7
A158 4F A2 CD C4 1F CD D0 1F : 5D
A160 21 52 A1 BE 20 02 18 F5 : 01
A168 77 21 9D A1 01 07 00 ED : CB
A170 B1 20 EA 79 FE 05 20 08 : 5F
A178 21 DA A0 35 E1 C3 38 A0 : 4C
----------------------------------
SUM: A3 5C 6E DB DC 19 D4 AD 8000

A180 FE 06 20 04 E1 C3 38 A0 : A4
A188 21 A3 A1 01 05 00 ED B1 : 09
A190 F5 3A A7 A1 B9 20 03 F1 : 44
A198 18 C3 F1 18 0D 52 42 4A : CF
A1A0 4B 4C 49 02 01 04 03 00 : EA
A1A8 00 00 32 A7 A1 E1 CD 1E : 46
A1B0 20 CD F1 1F CD 00 A2 DD : 49
A1B8 21 CD A1 11 02 00 3D 28 : 07
A1C0 04 DD 19 18 F9 DD 5E 00 : 46
A1C8 DD 56 01 D5 C9 39 A2 53 : 00
A1D0 A1 50 A2 63 A2 38 A0 3A : AA
A1D8 A8 A1 6F 3A A9 A1 67 CD : 70
A1E0 1E 20 3A A7 A1 FE 01 20 : DF
A1E8 02 3E 41 FE 02 20 02 3E : E1
A1F0 3E FE 03 20 02 3E 56 FE : F3
A1F8 04 20 02 3E 3C C3 F4 1F : 76
----------------------------------
SUM: 44 2C 11 24 0B 28 6D 84 FD26

A200 3A A7 A1 FE 01 20 01 25 : C7
A208 FE 02 20 01 2C FE 03 20 : 6E
A210 01 24 FE 04 20 01 2D CD : 42
A218 1B 20 FE 7B 20 03 3E 02 : 17
A220 C9 FE 2A 20 03 3E 03 C9 : 1E
A228 FE 40 20 03 3E 04 C9 FE : 6A
A230 4F 20 03 3E 05 C9 3E 01 : BD
A238 C9 7D 32 A8 A1 7C 32 A9 : 18
A240 A1 3E FF 32 4F A2 CD D7 : A5
A248 A1 CD C4 1F C3 AE A1 00 : 63
A250 7D 32 A8 A1 7C 32 A9 A1 : F0
A258 21 62 A2 35 CA 9C A2 C3 : 25
A260 53 A1 00 3A 4F A2 3C 20 : 7B
A268 03 C3 53 A1 E5 CD 00 A2 : 0E
A270 FE 01 20 02 18 0A FE 05 : 46
A278 20 02 18 18 E1 C3 53 A1 : EA
----------------------------------
SUM: 87 CE D4 A3 D9 03 F1 28 3665

A280 CD 1E 20 3E 40 CD F4 1F : 69
A288 E1 7D 32 A8 A1 7C 32 A9 : 30
A290 A1 C3 53 A1 CD 1E 20 CD : 30
A298 F1 1F 18 EC 3A DA A0 FE : C6
A2A0 14 20 02 18 07 3C 32 DA : 9D
A2A8 A0 C3 38 A0 3E 0C CD F4 : 46
A2B0 1F CD E2 1F 20 2A 2A 2A : 8B
A2B8 2A 2A 2A 2A 2A 2A 2A 2A : 50
A2C0 2A 2A 2A 2A 2A 2A 2A 2A : 50
A2C8 0D 20 2A 20 20 20 20 20 : F7
A2D0 20 20 20 20 20 20 20 20 : 00
A2D8 20 20 20 20 2A 0D 20 2A : 01
A2E0 20 43 4F 4E 47 52 41 54 : 2E
A2E8 55 4C 41 54 49 4F 4E 53 : 6F
A2F0 20 2A 0D 20 2A 20 20 20 : 01
A2F8 20 20 20 20 20 20 20 20 : 00
----------------------------------
SUM: 69 BA 54 E0 E5 35 92 30 1C2D

A300 20 20 20 20 20 20 2A 0D : F7
A308 20 2A 2A 2A 2A 2A 2A 2A : 46
A310 2A 2A 2A 2A 2A 2A 2A 2A : 50
A318 2A 2A 2A 2A 0D 00 C9 20 : 9E
A320 20 20 20 20 20 20 20 20 : 00
A328 20 20 20 20 20 20 20 20 : 00
A330 20 20 20 20 20 20 20 20 : 00
A338 20 20 20 20 20 20 20 20 : 00
A340 20 20 20 20 20 20 20 20 : 00
A348 20 20 20 20 20 20 20 20 : 00
A350 20 20 20 20 20 20 20 2A : 0A
A358 2A 2A 2A 20 20 20 20 20 : 1E
A360 20 3E 2A 2A 2A 2A 2A 2A : 5A
A368 2A 2A 2A 2A 20 20 20 20 : 28
A370 20 20 20 20 20 20 20 2A : 0A
A378 2A 2A 2A 20 20 20 20 20 : 1E
----------------------------------
SUM: 32 5A 46 32 0B FE D1 1F 2817

A380 20 20 20 20 20 20 20 20 : 00
A388 20 20 20 20 20 20 20 20 : 00
A390 20 20 20 20 20 20 20 20 : 00
A398 20 20 20 20 20 20 20 20 : 00
A3A0 20 20 20 20 20 20 20 20 : 00
A3A8 20 20 20 20 20 20 20 20 : 00
A3B0 20 20 20 20 20 20 20 20 : 00
A3B8 20 20 20 20 20 20 20 20 : 00
A3C0 20 20 20 20 20 20 20 20 : 00
A3C8 20 20 20 20 20 20 20 20 : 00
A3D0 20 20 20 20 20 20 20 20 : 00
A3D8 20 20 20 20 20 20 20 20 : 00
A3E0 20 20 20 20 20 20 2A 20 : 0A
A3E8 20 20 2A 20 20 20 20 20 : 0A
A3F0 20 20 20 20 20 2A 20 2A : 14
A3F8 20 20 20 20 20 20 20 20 : 00
----------------------------------
SUM: 00 00 0A 00 00 0A 0A 0A 526B

A400 20 20 20 20 2A 20 20 20 : 0A
A408 2A 20 20 20 20 20 20 20 : 0A
A410 20 20 20 2A 20 20 20 20 : 0A
A418 20 2A 20 20 20 20 20 20 : 0A
A420 20 20 2A 2A 2A 2A 2A 2A : 3C
A428 2A 2A 2A 20 20 20 20 20 : 1E
A430 3E 20 20 2A 20 20 20 20 : 28
A438 20 2A 20 20 20 20 20 20 : 0A
A440 20 20 20 20 20 20 20 20 : 00
A448 20 20 20 20 20 20 20 20 : 00
A450 20 20 20 20 20 20 20 20 : 00
A458 20 20 20 20 20 20 20 2A : 0A
A460 2A 2A 2A 2A 2A 2A 2A 2A : 50
A468 2A 2A 2A 2A 2A 2A 2A 2A : 50
A470 2A 2A 2A 2A 2A 2A 2A 2A : 50
A478 2A 2A 2A 2A 2A 2A 2A 20 : 46
----------------------------------
SUM: 5A 46 3C 46 3C 32 32 32 FD3E

A480 40 40 40 40 20 40 40 40 : E0
A488 40 40 20 40 40 40 40 20 : C0
A490 20 40 40 40 40 20 40 40 : C0
A498 40 2A 20 2A 40 40 40 20 : 94
A4A0 20 20 40 40 40 40 20 40 : A0
A4A8 2A 20 2A 40 40 40 40 20 : 94
A4B0 20 40 20 40 40 40 40 4F : CF
A4B8 20 2A 40 40 40 40 3E 20 : A8
A4C0 40 40 40 20 40 40 40 40 : E0
A4C8 20 40 40 40 40 20 20 40 : A0
A4D0 40 40 40 40 20 40 40 40 : E0
A4D8 40 40 40 40 20 20 20 2A : 8A
A4E0 2A 2A 2A 2A 2A 2A 2A 2A : 50
A4E8 2A 2A 2A 2A 2A 2A 2A 2A : 50
A4F0 2A 2A 2A 2A 2A 2A 2A 2A : 50
A4F8 2A 2A 2A 2A 2A 2A 2A 20 : 46
----------------------------------
SUM: F2 3C 32 72 48 48 46 17 4BDE

A500 20 20 20 20 20 20 20 20 : 00
A508 20 20 20 20 20 20 20 20 : 00
A510 20 20 20 20 20 20 20 20 : 00
A518 20 20 20 20 20 20 20 20 : 00
A520 3E 20 20 2A 2A 2A 2A 20 : 46
A528 20 20 20 20 20 20 20 20 : 00
A530 20 20 20 20 2A 2A 2A 2A : 28
A538 20 20 20 20 20 20 20 20 : 00
A540 20 20 20 20 20 2A 2A 2A : 1E
A548 2A 20 20 20 20 20 20 20 : 0A
A550 20 20 20 20 20 20 2A 2A : 14
A558 2A 2A 20 20 20 20 20 20 : 14
A560 20 20 20 20 20 20 20 2A : 0A
A568 2A 2A 2A 20 20 20 20 20 : 1E
A570 20 20 20 20 20 20 20 20 : 00
A578 20 20 20 20 20 20 20 20 : 00
----------------------------------
SUM: 3C 14 0A 0A 14 1E 28 28 572C

A580 20 20 20 20 20 20 20 20 : 00
A588 20 20 20 20 20 20 20 20 : 00
A590 20 20 20 20 20 20 20 20 : 00
A598 20 20 20 20 20 20 20 20 : 00
A5A0 20 20 20 20 20 20 20 20 : 00
A5A8 20 20 20 20 20 20 20 20 : 00
A5B0 20 20 20 20 20 20 20 20 : 00
A5B8 20 20 20 20 20 20 20 20 : 00
A5C0 7B 40 7B 7B 3E 7B 7B 20 : 05
A5C8 7B 7B 7B 7B 20 7B 20 20 : C7
A5D0 7B 20 20 7B 20 20 7B 20 : 11
A5D8 20 20 20 2A 7B 2A 20 20 : 6F
A5E0 7B 20 20 7B 20 20 7B 20 : 11
A5E8 20 20 2A 7B 2A 20 20 20 : 6F
A5F0 7B 20 20 7B 20 20 7B 20 : 11
A5F8 20 2A 7B 2A 20 20 20 20 : 6F
----------------------------------
SUM: C7 85 1B 36 83 C0 6C 00 30DB

A600 7B 20 20 7B 20 20 7B 20 : 11
A608 2A 7B 2A 20 20 20 20 20 : 6F
A610 7B 40 20 7B 40 20 7B 20 : 51
A618 7B 20 7B 7B 7B 7B 20 20 : C7
A620 20 20 20 20 20 20 20 20 : 00
A628 20 20 20 20 20 20 20 20 : 00
A630 20 20 20 20 20 20 20 20 : 00
A638 20 20 20 20 20 20 20 20 : 00
A640 20 20 20 20 20 20 20 20 : 00
A648 20 20 20 20 20 20 20 3E : 1E
A650 20 20 20 20 20 20 2A 2A : 14
A658 2A 20 20 20 20 20 20 20 : 0A
A660 20 20 20 20 20 20 20 20 : 00
A668 2A 20 20 2A 20 20 20 20 : 14
A670 20 20 20 20 20 20 20 20 : 00
A678 2A 20 20 2A 20 20 20 20 : 14
----------------------------------
SUM: 39 7B 65 25 7B 5B C0 28 68F8

A680 20 20 20 20 20 2A 2A 2A : 1E
A688 2A 2A 2A 2A 20 20 20 20 : 28
A690 20 20 20 20 20 2A 20 20 : 0A
A698 2A 20 20 20 20 20 20 20 : 0A
A6A0 20 20 20 20 20 2A 20 20 : 0A
A6A8 2A 20 20 20 20 20 20 20 : 0A
A6B0 20 20 20 20 20 20 20 20 : 00
A6B8 2A 2A 2A 20 20 20 20 20 : 1E
A6C0 20 20 20 20 20 20 20 20 : 00
A6C8 20 20 20 20 20 20 20 20 : 00
A6D0 20 20 20 20 20 20 20 20 : 00
A6D8 20 20 20 20 20 20 20 4F : 2F
A6E0 20 20 20 20 20 20 20 20 : 00
A6E8 20 20 20 20 20 20 4F 20 : 2F
A6F0 20 20 20 4F 2A 2A 4F 2A : 7C
A6F8 2A 4F 20 20 20 20 20 20 : 39
----------------------------------
SUM: 32 43 14 39 0A 28 68 43 6D3B

A700 20 20 20 2A 20 20 2A 20 : 14
A708 20 2A 20 20 20 20 20 20 : 0A
A710 20 20 20 2A 20 20 2A 20 : 14
A718 20 2A 20 20 20 20 20 20 : 0A
A720 20 20 20 4F 2A 2A 4F 2A : 7C
A728 2A 4F 20 20 20 20 20 20 : 39
A730 20 20 20 2A 3E 40 2A 20 : 52
A738 20 2A 20 20 20 20 20 20 : 0A
A740 20 20 20 2A 20 20 2A 20 : 14
A748 20 2A 20 20 20 20 20 20 : 0A
A750 20 20 20 4F 2A 2A 4F 2A : 7C
A758 2A 4F 20 20 20 20 20 20 : 39
A760 20 20 20 20 20 20 20 20 : 00
A768 20 20 20 20 20 20 20 4F : 2F
A770 20 20 20 20 20 20 20 20 : 00
A778 20 20 20 20 20 20 4F 20 : 2F
----------------------------------
SUM: 14 86 00 86 32 34 B5 43 B68C

A780 20 20 20 20 20 20 20 20 : 00
A788 20 20 20 20 20 20 20 20 : 00
A790 20 20 20 20 20 20 20 20 : 00
A798 2A 20 20 20 20 20 20 20 : 0A
A7A0 20 20 20 20 20 20 2A 2A : 14
A7A8 2A 2A 2A 20 20 20 20 20 : 1E
A7B0 20 20 20 20 20 2A 2A 2A : 1E
A7B8 2A 2A 2A 2A 20 20 20 3E : 46
A7C0 20 20 20 20 2A 2A 2A 2A : 28
A7C8 2A 2A 2A 2A 2A 20 20 20 : 32
A7D0 20 20 20 20 2A 2A 2A 2A : 28
A7D8 2A 2A 2A 2A 2A 20 20 20 : 32
A7E0 20 20 20 20 7B 7B 7B 7B : 6C
A7E8 7B 7B 7B 7B 7B 20 20 20 : C7
A7F0 20 20 20 20 20 7B 7B 7B : 11
A7F8 7B 7B 7B 7B 20 20 20 20 : 6C
----------------------------------
SUM: E8 DE DE D4 DE D4 DE FC A79B

A800 20 20 20 20 20 20 7B 7B : B6
A808 7B 7B 7B 20 20 20 20 20 : 11
A810 20 20 20 20 20 20 20 7B : 5B
A818 7B 7B 20 20 20 20 20 20 : B6
A820 20 20 20 20 20 20 20 20 : 00
A828 20 20 20 20 20 20 20 3E : 1E
A830 20 20 20 20 20 2A 20 20 : 0A
A838 20 20 20 20 20 20 20 20 : 00
A840 20 20 20 20 20 2A 20 20 : 0A
A848 20 20 20 20 20 20 20 20 : 00
A850 20 20 20 2A 2A 2A 2A 2A : 32
A858 2A 20 20 20 20 20 20 20 : 0A
A860 20 20 20 20 20 2A 20 20 : 0A
A868 2A 20 20 20 20 20 20 20 : 0A
A870 20 20 20 20 20 2A 20 20 : 0A
A878 2A 20 20 20 20 20 20 20 : 0A
----------------------------------
SUM: D4 B6 5B 0A 0A 32 65 DE 1FC4

A880 20 20 20 20 20 2A 20 20 : 0A
A888 2A 20 20 20 20 20 20 20 : 0A
A890 20 20 20 20 20 2A 20 20 : 0A
A898 2A 20 2A 20 20 20 20 20 : 14
A8A0 20 20 20 20 2A 2A 20 20 : 14
A8A8 2A 2A 2A 20 20 20 20 20 : 1E
A8B0 20 20 20 20 20 20 20 20 : 00
A8B8 20 20 20 20 20 20 20 3E : 1E
A8C0 20 20 20 20 20 20 20 20 : 00
A8C8 20 20 20 20 20 20 20 20 : 00
A8D0 20 20 20 20 20 20 20 20 : 00
A8D8 20 20 20 20 20 20 20 20 : 00
A8E0 40 20 40 20 40 20 40 20 : 80
A8E8 40 20 40 20 40 20 40 40 : A0
A8F0 20 40 20 40 20 40 20 40 : 80
A8F8 20 40 20 40 20 40 20 4F : 8F
----------------------------------
SUM: 5E 4A 54 40 4A 5E 40 8D 581A

A900 4F 4F 4F 4F 4F 4F 4F 4F : 78
A908 4F 4F 4F 4F 4F 4F 4F 4F : 78
A910 4F 4F 4F 4F 4F 4F 4F 4F : 78
A918 4F 4F 4F 4F 4F 4F 4F 4F : 78
A920 4F 4F 4F 4F 4F 4F 4F 4F : 78
A928 4F 4F 4F 4F 4F 4F 4F 4F : 78
A930 4F 4F 4F 4F 4F 4F 4F 4F : 78
```

```
A938 4F 4F 4F 4F 4F 4F 4F 4F : 78
A940 4F 4F 4F 4F 4F 4F 4F 4F : 78
A948 4F 4F 4F 4F 4F 4F 4F 2A : 53
A950 2A 2A 2A 2A 2A 2A 2A 2A : 50
A958 2A 2A 2A 2A 2A 2A 2A 4F : 75
A960 4F 4F 4F 4F 4F 4F 4F 4F : 78
A968 4F 4F 4F 4F 4F 4F 4F 4F : 78
A970 3E 40 20 20 20 20 20 20 : 3E
A978 20 20 20 20 20 20 4F 4F : 5E
---------------------------------
SUM: 66 68 48 48 48 48 77 77 EA8A

A980 20 20 20 20 20 20 20 20 : 00
A988 20 20 20 20 20 20 4F 4F : 5E
A990 20 20 20 20 20 20 20 20 : 00
A998 2A 20 20 20 20 20 4F 4F : 68
A9A0 20 20 20 20 20 20 20 2A : 0A
A9A8 20 2A 2A 20 20 20 4F 4F : 72
A9B0 20 20 20 20 20 20 20 2A : 0A
A9B8 20 2A 20 20 20 20 4F 4F : 68
A9C0 20 20 20 20 20 20 2A 20 : 0A
A9C8 2A 20 2A 2A 20 20 4F 4F : 72
A9D0 20 20 20 20 20 20 2A 20 : 0A
A9D8 2A 20 2A 20 20 20 4F 4F : 72
A9E0 20 20 20 20 20 20 20 2A : 0A
A9E8 20 2A 20 20 20 20 4F 4F : 68
A9F0 4F 4F 4F 4F 4F 4F 4F 4F : 78
A9F8 4F 4F 4F 4F 4F 4F 4F 4F : 78
---------------------------------
SUM: 7C 7C 7C 5E 5E 5E BB C5 576B

AA00 4F 4F 4F 4F 4F 4F 4F 4F : 78
AA08 4F 4F 4F 4F 4F 4F 4F 4F : 78
AA10 20 20 20 20 20 20 20 20 : 00
AA18 20 20 20 20 20 20 4F 4F : 5E
AA20 20 20 20 40 40 40 40 40 : A0
AA28 40 40 20 20 20 20 4F 4F : 9E
AA30 20 20 20 40 20 20 20 20 : 20
AA38 20 40 20 20 20 20 4F 4F : 7E
AA40 20 20 20 40 20 2A 4F 3E : 77
AA48 20 40 20 20 20 20 4F 4F : 7E
AA50 20 20 20 40 20 20 20 20 : 20
AA58 20 40 20 20 20 20 4F 4F : 7E
AA60 20 20 20 40 40 40 40 40 : A0
AA68 40 40 20 20 20 20 4F 4F : 9E
AA70 20 20 20 20 20 20 20 20 : 00
AA78 20 20 20 20 20 20 4F 4F : 5E
---------------------------------
SUM: 9E FE 5E FE 9E A8 16 05 09A9

AA80 20 20 20 20 20 20 20 20 : 00
AA88 20 20 20 20 20 20 4F 4F : 5E
AA90 4F 4F 4F 4F 4F 4F 4F 4F : 78
AA98 4F 4F 4F 4F 4F 4F 4F 4F : 78
AAA0 40 4F 40 4F 40 4F 40 4F : 3C
AAA8 40 4F 40 4F 40 4F 40 40 : 2D
AAB0 4F 40 4F 40 4F 40 4F 40 : 3C
AAB8 4F 40 4F 40 4F 40 4F 4F : 4B
AAC0 40 4F 40 4F 40 4F 40 4F : 3C
AAC8 40 4F 40 4F 2A 4F 40 40 : 17
AAD0 4F 40 4F 40 4F 2A 4F 40 : 26
AAD8 4F 40 4F 40 4F 40 4F 4F : 4B
AAE0 40 4F 2A 4F 40 4F 40 4F : 26
AAE8 40 4F 40 4F 40 4F 40 40 : 2D
AAF0 4F 40 4F 40 4F 40 4F 2A : 26
AAF8 4F 40 4F 40 4F 40 4F 4F : 4B
---------------------------------
SUM: 38 38 22 38 22 22 67 51 C457

AB00 2A 4F 40 4F 40 4F 40 4F : 26
AB08 40 4F 40 4F 40 4F 40 40 : 2D
AB10 4F 40 4F 40 4F 40 4F 40 : 3C
AB18 4F 40 4F 2A 4F 40 4F 4F : 35
AB20 40 4F 40 4F 40 4F 2A 4F : 26
AB28 40 4F 40 4F 40 4F 40 40 : 2D
AB30 4F 40 4F 40 4F 40 4F 40 : 3C
AB38 4F 40 4F 40 4F 40 3E 40 : 2B
AB40 20 40 2A 40 20 40 2A 40 : 94
AB48 20 40 2A 40 20 40 2A 40 : 94
AB50 20 40 20 40 20 40 20 40 : 80
AB58 20 40 20 40 20 40 20 40 : 80
AB60 20 40 20 40 20 40 20 40 : 80
AB68 20 40 20 40 20 40 20 3E : 7E
AB70 20 40 20 20 20 20 20 20 : 20
AB78 20 20 20 20 20 20 20 20 : 00
---------------------------------
SUM: 26 1C 50 E6 3C FC 29 EB 07B9

AB80 40 20 40 20 40 20 40 20 : 80
AB88 40 20 40 20 40 20 40 20 : 80
AB90 40 20 40 20 40 20 40 20 : 80
AB98 40 20 40 20 40 20 40 20 : 80
ABA0 40 20 40 20 40 20 40 20 : 80
ABA8 40 20 40 20 40 20 40 20 : 80
ABB0 40 20 40 20 40 20 40 20 : 80
ABB8 40 20 40 20 40 20 40 20 : 80
ABC0 40 20 40 20 40 20 40 20 : 80
ABC8 40 20 40 20 40 20 40 20 : 80
ABD0 40 20 40 20 40 20 40 20 : 80
ABD8 40 20 40 20 40 20 40 4F : AF
ABE0 4F 4F 4F 4F 4F 4F 4F 4F : 78
ABE8 4F 4F 4F 4F 4F 4F 4F 4F : 78
ABF0 3E 40 20 20 20 20 20 20 : 3E
ABF8 20 20 20 20 20 20 4F 4F : 5E
---------------------------------
SUM: FC 7E DE 5E DE 5E 0D BC D50D

AC00 20 20 20 20 20 20 20 20 : 00
AC08 20 20 20 20 20 20 4F 4F : 5E
AC10 20 20 20 20 20 2A 2A 2A : 1E
AC18 2A 2A 20 20 20 20 4F 4F : 72
AC20 20 20 20 20 20 20 2A 20 : 0A
AC28 20 2A 20 20 20 20 4F 4F : 68
AC30 20 20 20 20 20 20 2A 20 : 0A
AC38 20 2A 20 20 20 20 4F 4F : 68
AC40 20 20 20 20 20 20 2A 2A : 14
AC48 2A 2A 20 20 20 20 4F 4F : 72
AC50 20 20 20 20 20 20 20 20 : 00
AC58 20 2A 20 20 20 20 4F 4F : 68
AC60 20 20 20 20 20 20 20 20 : 00
AC68 20 20 20 20 20 20 4F 4F : 5E
AC70 4F 4F 4F 4F 4F 4F 4F 4F : 78
AC78 4F 4F 4F 4F 4F 4F 4F 20 : 49
---------------------------------
SUM: 72 90 5E 5E 5E 68 CF 8C AF44

AC80 20 20 20 7B 20 20 2A 20 : 65
AC88 20 20 20 20 20 20 20 20 : 00
AC90 7B 20 20 20 20 20 20 4F : 8A
AC98 20 4F 20 7B 2A 4F 20 7B : 1E
ACA0 20 2A 20 20 20 20 20 20 : 0A
ACA8 20 20 20 7B 20 20 20 20 : 5B
ACB0 2A 20 20 20 20 20 20 20 : 0A
ACB8 20 20 20 20 20 7B 20 20 : 5B
ACC0 20 3E 20 20 20 2A 20 4F : 57
ACC8 20 2A 20 20 20 20 7B 20 : 65
ACD0 20 20 4F 20 20 4F 20 20 : 5E
ACD8 20 20 20 20 4F 20 20 4F : 5E
ACE0 20 20 20 20 20 2A 20 2A : 14
ACE8 20 2A 20 20 20 20 20 20 : 0A
ACF0 2A 20 20 20 20 20 2A 20 : 14
ACF8 20 4F 7B 7B 20 20 20 20 : E5
---------------------------------
SUM: 6F 9A 8A 6C 68 9E 6F F2 540C

AD00 20 20 7B 20 20 20 20 20 : 5B
AD08 2A 20 20 20 20 20 20 20 : 0A
AD10 20 20 20 20 4F 20 20 7B : 8A
AD18 7B 20 20 20 20 7B 20 20 : B6
AD20 20 40 20 20 40 20 20 20 : 40
AD28 20 20 20 20 2A 20 20 2A : 14
AD30 7B 7B 7B 7B 20 20 20 20 : 6C
AD38 20 20 20 20 20 20 20 20 : 00
AD40 20 7B 7B 20 20 20 7B 7B : 6C
AD48 7B 7B 7B 7B 7B 20 20 20 : C7
AD50 20 7B 7B 20 20 20 7B 20 : 11
AD58 20 20 20 2A 20 20 20 20 : 0A
AD60 20 7B 7B 20 20 2A 7B 20 : 1B
AD68 40 20 20 20 20 20 20 20 : 20
AD70 20 7B 7B 20 20 20 7B 7B : 6C
AD78 7B 7B 7B 7B 7B 20 20 20 : C7
---------------------------------
SUM: 96 9D D8 1B 0F 65 6C 1B EA9B

AD80 20 7B 7B 20 20 20 7B 7B : 6C
AD88 7B 20 20 20 20 20 20 20 : 5B
AD90 7B 7B 7B 7B 20 20 7B 7B : 22
AD98 7B 20 40 20 20 3E 20 4F : C8
ADA0 20 20 20 20 20 20 7B 7B : B6
ADA8 7B 20 20 20 20 20 20 20 : 5B
ADB0 20 2A 20 20 20 20 2A 20 : 14
ADB8 20 20 20 20 40 20 40 4F : 6F
ADC0 20 20 20 20 20 20 20 20 : 00
ADC8 20 20 20 20 20 20 4F 20 : 2F
ADD0 20 20 20 20 20 20 20 20 : 00
ADD8 20 20 20 20 20 20 20 20 : 00
ADE0 20 4F 2A 2A 2A 4F 4F 2A : B5
ADE8 2A 2A 2A 2A 2A 20 20 20 : 32
ADF0 20 4F 2A 4F 2A 2A 2A 2A : 90
ADF8 4F 2A 2A 4F 2A 20 20 20 : 7C
---------------------------------
SUM: A5 32 FE CD 48 57 A3 83 444C

AE00 20 4F 2A 2A 2A 2A 2A 2A : 6B
AE08 4F 2A 2A 2A 2A 20 20 20 : 57
AE10 20 2A 2A 2A 2A 2A 4F 2A : 6B
AE18 2A 2A 2A 4F 4F 20 20 20 : 7C
AE20 20 2A 2A 2A 2A 2A 2A 2A : 46
AE28 2A 2A 2A 2A 4F 20 20 20 : 57
AE30 20 3E 2A 2A 4F 4F 2A 2A : A4
AE38 4F 2A 2A 2A 4F 20 20 20 : 7C
AE40 20 20 20 20 20 20 20 20 : 00
AE48 20 20 20 20 20 20 20 4F : 2F
AE50 20 20 20 20 20 20 20 20 : 00
AE58 20 20 20 20 20 20 4F 20 : 2F
AE60 7B 20 20 20 20 20 20 20 : 5B
AE68 20 20 20 20 20 20 20 2A : 0A
AE70 20 20 20 20 20 20 20 20 : 00
AE78 20 20 20 20 20 20 20 20 : 00
---------------------------------
SUM: CD 89 50 75 E4 4D 7C 61 B0B1

AE80 7B 20 40 20 20 4F 4F 4F : 08
AE88 4F 4F 20 20 40 20 20 7B : D9
AE90 7B 20 20 40 20 4F 4F 4F : 08
AE98 4F 4F 20 40 20 20 20 20 : 7E
AEA0 20 20 20 20 40 4F 4F 2A : 88
AEA8 4F 4F 40 20 20 20 20 20 : 7E
AEB0 20 20 20 40 20 4F 4F 4F : AD
AEB8 4F 4F 20 40 20 20 20 20 : 7E
AEC0 20 20 40 20 20 4F 4F 4F : AD
AEC8 4F 4F 20 20 40 20 20 20 : 7E
AED0 20 20 20 20 20 20 20 20 : 00
AED8 20 20 20 20 20 20 20 20 : 00
AEE0 20 20 20 20 20 20 20 20 : 00
AEE8 20 20 20 20 20 20 20 20 : 00
AEF0 20 20 20 20 20 20 20 20 : 00
AEF8 20 20 20 20 20 20 3E 20 : 1E
---------------------------------
SUM: A1 EB 60 80 60 EB 09 21 3B87

AF00 20 20 4F 7B 20 4F 20 20 : B9
AF08 20 20 4F 20 20 4F 20 20 : 5E
AF10 20 20 20 20 20 20 20 4F : 2F
AF18 20 7B 20 2A 20 2A 20 7B : CA
AF20 20 20 20 4F 20 40 2A 20 : 59
AF28 20 20 4F 20 20 20 40 20 : 4F
AF30 40 4F 2A 20 2A 20 20 40 : 83
AF38 20 40 2A 40 4F 4F 20 20 : A8
AF40 4F 20 40 2A 20 3E 20 2A : 81
AF48 2A 2A 20 20 20 20 20 4F : 43
AF50 20 20 2A 20 40 20 20 20 : 2A
AF58 20 20 20 4F 20 40 4F 20 : 7E
AF60 2A 40 4F 40 4F 7B 20 40 : 23
AF68 20 40 20 20 2A 2A 20 20 : 34
AF70 20 4F 20 20 20 20 4F 20 : 5E
AF78 20 2A 20 20 20 7B 20 20 : 65
---------------------------------
SUM: 63 5C CB 0D 92 B5 88 03 DC33

AF80 4F 20 20 40 2A 20 20 20 : 59
AF88 40 2A 4F 20 20 20 20 20 : 59
AF90 20 20 20 20 4F 20 20 4F : 5E
AF98 20 20 20 20 4F 20 40    : 2F
---------------------------------
SUM: CF 8A AF A0 E8 80 A0 8F 5714
```

リスト2　碁石拾いソースリスト

```
A000                1          START   $A000
A000                2 #PRINT   EQU $1FF4
A000                3 #PRINTS  EQU $1FF1
A000                4 #MSX     EQU $1FE5
A000                5 #MPRINT  EQU $1FE2
A000                6 #GETKY   EQU $1FD0
A000                7 #BELL    EQU $1FC4
A000                8 #SCRN    EQU $201B
A000                9 #LOC     EQU $201E
A000               10 #WIDTH   EQU $2030
A000               11 ;
A000               12 ;
A000               13 ;
A000 3E 01         14          LD      A,1
A002 32 DA A0      15          LD      (ROUND),A
A005 3E 0C         16          LD      A,$0C
A007 CD F4 1F      17          CALL    #PRINT
A00A               18
A00A 11 DB A0      19          LD      DE,SCREEN
A00D 06 0B         20          LD      B,11
A00F               21 LOOP
A00F 1A            22          LD      A,(DE)
A010 6F            23          LD      L,A
A011 13            24          INC     DE
A012 1A            25          LD      A,(DE)
A013 67            26          LD      H,A
A014 13            27          INC     DE
A015 CD 1E 20      28          CALL    #LOC
A018               29
A018 CD E5 1F      30          CALL    #MSX
A01B 21 09 00      31          LD      HL,9
A01E 19            32          ADD     HL,DE
A01F 54 5D         33          LD      DE,HL
A021 10 EC         34          DJNZ    LOOP
A023               35
A023 21 03 0A      36          LD      HL,$0A03
A026 CD 1E 20      37          CALL    #LOC
A029 11 C7 A0      38          LD      DE,WALL
A02C CD E5 1F      39          CALL    #MSX
A02F 21 03 15      40          LD      HL,$1503
A032 CD 1E 20      41          CALL    #LOC
A035 CD E5 1F      42          CALL    #MSX
A038               43
A038               44
A038               45 NAKAMI
A038 3A DA A0      46          LD      A,(ROUND)
A03B 3D            47          DEC     A
A03C 3C 20 05      48          IF INC(A)=0 THEN  LD A,$0C  JP #PRINT
```


▶祝氏が「私は誰の挑戦でも受ける」といわなくなったことに疑問を持っているのは私だけではないはず。
佐藤　貴嗣（19）神奈川県

```
A03F 3E 0C C3
A042 F4 1F
A044 06 00       49          LD      B,0
A046 0E 0A       50          LD      C,10
A048 16 30       51          LD      D,$30
A04A             52 JUSSHIN
A04A FE 0A       53          CP      10
A04C 38 04       54          JR      C,ROUNDHYOJI
A04E 04          55          INC     B
A04F 91          56          SUB     C
A050 18 F8       57          JR      JUSSHIN
A052             58 ROUNDHYOJI
A052 21 11 07    59          LD      HL,$0711
A055 CD 1E 20    60          CALL    #LOC
A058 82          61          ADD     A,D
A059 CD F4 1F    62          CALL    #PRINT
A05C 3E 1D       63          LD      A,$1D
A05E CD F4 1F    64          CALL    #PRINT
A061 CD F4 1F    65          CALL    #PRINT
A064 78          66          LD      A,B
A065 82          67          ADD     A,D
A066 CD F4 1F    68          CALL    #PRINT
A069             69
A069             70
A069 AF          71          XOR     A
A06A 32 62 A2    72          LD      (ISHIKAZU),A
A06D 3A DA A0    73          LD      A,(ROUND)
A070 47          74          LD      B,A
A071 21 7F A2    75          LD      HL,ROUNDDATA-160
A074 11 A0 00    76          LD      DE,16*10
A077             77 LOOP2
A077 19          78          ADD     HL,DE
A078 10 FD       79          DJNZ    LOOP2
A07A             80
A07A 44 4D       81          LD      BC,HL
A07C 21 00 0B    82          LD      HL,$0B00
A07F CD 1E 20    83          CALL    #LOC
A082 26 0B       84          LD      H,11
A084             85 PUT1
A084 2E 04       86          LD      L,4
A086 CD E2 1F    87          CALL    #MPRINT
A089 20 20 20    88          DM "   ■" DB 0
A08C 7B 00
A08E             89 PUT2
A08E 0A          90          LD      A,(BC)
A08F 03          91          INC     BC
A090 CD F4 1F    92          CALL    #PRINT
A093 FE 2A 20    93          IF A="*" THEN PUSH HL  LD HL,ISHIKAZU  INC (HL)  POP
 HL
A096 06 E5 21
A099 62 A2 34
A09C E1
A09D FE 3E 20    94          IF A=">" THEN LD A,L  LD (XZAHYO),A  LD A,H  LD (YZA
HYO),A
A0A0 08 7D 32
A0A3 A8 A1 7C
A0A6 32 A9 A1
A0A9 2C          95          INC     L
A0AA 7D          96          LD      A,L
A0AB FE 14 28    97          IF A<>20 THEN JR PUT2
A0AE 02 18 DD
A0B1 CD E2 1F    98          CALL    #MPRINT
A0B4 7B 0D 00    99          DB "■",$0D,0
A0B7 24         100          INC     H
A0B8 7C         101          LD      A,H
A0B9 FE 15 28   102          IF A<>21 THEN JR PUT1
A0BC 02 18 C5
A0BF            103
A0BF 3E 02      104          LD      A,2
A0C1 32 A7 A1   105          LD      (HOKO),A
A0C4 C3 53 A1   106          JP      KEY
A0C7            107
A0C7            108
A0C7 7B 7B 7B   109 WALL     DM "■■■■■■■■■■■■■■■■■■" DB 0
A0CA 7B 7B 7B
A0CD 7B 7B 7B
A0D0 7B 7B 7B
A0D3 7B 7B 7B
A0D6 7B 7B 7B
A0D9 00
A0DA            110
A0DA 00         111 ROUND    DS 1
A0DB            112
A0DB            113 SCREEN
A0DB 04 04 BA   114          DB    4,4 DM "ｺﾞｲｼ ﾋﾛｲ" DB 0
A0DE DE B2 BC
A0E1 20 CB DB
A0E4 B2 00
A0E6 0A 07 52   115          DB  10,7 DM "ROUND:  " DB 0
A0E9 4F 55 4E
A0EC 44 3A 20
A0EF 20 00
A0F1 19 06 3E   116          DB  25,6 DM "> TAKER " DB 0
A0F4 20 54 41
A0F7 4B 45 52
A0FA 20 00
A0FC 19 08 2A   117          DB  25,8 DM "* GOISHI" DB 0
A0FF 20 47 4F
A102 49 53 48
A105 49 00
A107 19 0A 7B   118          DB 25,10 DM "■ WALL  " DB 0
A10A 20 57 41
A10D 4C 4C 20
A110 20 00
A112 19 0C 40   119          DB 25,12 DM "@ STONE " DB 0
A115 20 53 54
A118 4F 4E 45
A11B 20 00
A11D 19 0E 4F   120          DB 25,14 DM "O HOLE  " DB 0
A120 20 48 4F
A123 4C 45 20
A126 20 00
A128 1A 11 49   121          DB 26,17,"I",$1F,$1F,$1F,$1F,$1D,"K",$20,0
A12B 1F 1F 1F
A12E 1F 1D 4B
A131 20 00
A133 18 13 4A   122          DB 24,19 DM "J > L R " DB 0
A136 20 3E 20
A139 4C 20 52
A13C 20 00
A13E 1F 13 3A   123          DB 31,19 DM ":REPLACE" DB 0
A141 52 45 50
A144 4C 41 43
A147 45 00
A149 1E 15 42   124          DB 30,21 DM "B:BACK"   DB 0
A14C 3A 42 41
A14F 43 4B 00
A152            125
A152 00         126 KEYBUF   DS 1
A153            127
A153            128
A153            129 KEY
A153 CD D7 A1   130          CALL    HYOJI
A156 E5         131          PUSH    HL
A157 32 4F A2   132          LD      (IKIOI),A
A15A CD C4 1F   133          CALL    #BELL
A15D            134 KEY2
A15D CD D0 1F   135          CALL    #GETKY
A160 21 52 A1   136          LD      HL,KEYBUF
A163 BE 20 02   137          IF A=(HL) THEN  JR KEY2
A166 18 F5
A168 77         138          LD      (HL),A
A169 21 9D A1   139          LD      HL,KEYDATA1
A16C 01 07 00   140          LD      BC,7
A16F ED B1      141          CPIR
A171 20 EA      142          JR      NZ,KEY2
A173 79         143          LD      A,C
A174 FE 05 20   144          IF A=5 THEN LD HL,ROUND  DEC (HL)  POP HL  JP NAKAMI
A177 08 21 DA
A17A A0 35 E1
A17D C3 38 A0
A180 FE 06 20   145          IF A=6 THEN POP HL JP NAKAMI
A183 04 E1 C3
A186 38 A0
A188 21 A3 A1   146          LD      HL,KEYDATA2
A18B 01 05 00   147          LD      BC,5
A18E ED B1      148          CPIR
A190 F5         149          PUSH    AF
A191 3A A7 A1   150          LD      A,(HOKO)
A194 B9 20 03   151          IF A=C THEN POP AF:JR KEY2
A197 F1 18 C3
A19A F1         152          POP     AF
A19B 18 0D      153          JR      MAIN
A19D            154
A19D 52 42 4A   155 KEYDATA1         DM "RBJKLI"
A1A0 4B 4C 49
A1A3 02 01 04   156 KEYDATA2         DB 2,1,4,3
A1A6 03
A1A7            157
A1A7 00         158 HOKO     DS    1
A1A8 00         159 XZAHYO   DS    1
A1A9 00         160 YZAHYO   DS    1
A1AA            161
A1AA            162
A1AA            163 MAIN
A1AA 32 A7 A1   164          LD      (HOKO),A
A1AD E1         165          POP     HL
A1AE            166 MAIN2
A1AE CD 1E 20   167          CALL    #LOC
A1B1 CD F1 1F   168          CALL    #PRINTS
A1B4 CD 00 A2   169          CALL    MOVE
A1B7            170
A1B7 DD 21 CD   171          LD      IX,JPTBL
A1BA A1
A1BB 11 02 00   172          LD      DE,2
A1BE            173 ADDJP
A1BE 3D         174          DEC     A
A1BF 28 04      175          JR      Z,JUMP
A1C1 DD 19      176          ADD     IX,DE
A1C3 18 F9      177          JR      ADDJP
A1C5            178 JUMP
A1C5 DD 5E 00   179          LD      E,(IX)
A1C8 DD 56 01   180          LD      D,(IX+1)
A1CB D5         181          PUSH    DE
A1CC C9         182          RET
A1CD            183 JPTBL
A1CD 39 A2 53   184          DEFW    SPACE,KEY,GOISHI,ISHI,NAKAMI
A1D0 A1 50 A2
A1D3 63 A2 38
A1D6 A0
A1D7            185
A1D7            186
A1D7            187 HYOJI
A1D7 3A A8 A1   188          LD      A,(XZAHYO):LD L,A
A1DA 6F
A1DB 3A A9 A1   189          LD      A,(YZAHYO):LD H,A
A1DE 67
A1DF CD 1E 20   190          CALL    #LOC
A1E2 3A A7 A1   191          LD      A,(HOKO)
A1E5 FE 01 20   192          IF A=1 THEN LD A,"A"
A1E8 02 3E 41
A1EB FE 02 20   193          IF A=2 THEN LD A,">"
A1EE 02 3E 3E
A1F1 FE 03 20   194          IF A=3 THEN LD A,"V"
A1F4 02 3E 56
A1F7 FE 04 20   195          IF A=4 THEN LD A,"<"
A1FA 02 3E 3C
A1FD C3 F4 1F   196          JP      #PRINT
A200            197
A200            198
A200            199 MOVE
A200 3A A7 A1   200          LD      A,(HOKO)
A203 FE 01 20   201          IF A=1 THEN DEC H
A206 01 25
A208 FE 02 20   202          IF A=2 THEN INC L
A20B 01 2C
A20D FE 03 20   203          IF A=3 THEN INC H
A210 01 24
A212 FE 04 20   204          IF A=4 THEN DEC L
A215 01 2D
A217            205
A217 CD 1B 20   206          CALL    #SCRN
A21A FE 7B 20   207          IF A="■" THEN LD A,2:RET
A21D 03 3E 02
A220 C9
A221 FE 2A 20   208          IF A="*" THEN LD A,3:RET
A224 03 3E 03
A227 C9
A228 FE 40 20   209          IF A="@" THEN LD A,4:RET
A22B 03 3E 04
A22E C9
A22F FE 4F 20   210          IF A="O" THEN LD A,5:RET
A232 03 3E 05
A235 C9
A236 3E 01 C9   211          LD A,1:RET
A239            212
A239            213
A239            214
A239            215 SPACE
A239 7D 32 A8   216          LD A,L:LD (XZAHYO),A
A23C A1
A23D 7C 32 A9   217          LD A,H:LD (YZAHYO),A
A240 A1
A241 3E FF      218          LD      A,$FF
A243 32 4F A2   219          LD      (IKIOI),A
A246 CD D7 A1   220          CALL    HYOJI
A249 CD C4 1F   221          CALL    #BELL
A24C C3 AE A1   222          JP      MAIN2
A24F            223
A24F 00         224 IKIOI    DS 1
A250            225
A250            226
A250            227 GOISHI
A250 7D 32 A8   228          LD A,L:LD (XZAHYO),A
A253 A1
A254 7C 32 A9   229          LD A,H:LD (YZAHYO),A
A257 A1
A258 21 62 A2   230          LD      HL,ISHIKAZU
A25B 35         231          DEC     (HL)
A25C CA 9C A2   232          JP      Z,CLEAR
```

▶ビデオプリンタなるものが出現したが，やはり画面をカメラで撮影したほうが色の差異もなくランニングコストも安くてすむような気がします。　北御門　恭子（27）大阪府

```
A25F C3 53 A1    233            JP      KEY
A262             234
A262 00          235 ISHIKAZU           DS 1
A263             236
A263             237
A263             238 ISHI
A263 3A 4F A2    239            LD      A,(IKIOI)
A266 3C 20 03    240            IF INC(A)=0 THEN JP KEY
A269 C3 53 A1
A26C E5          241            PUSH    HL
A26D CD 00 A2    242            CALL    MOVE
A270 FE 01 20    243            IF A=1 THEN  JR MOVESTONE
A273 02 18 0A
A276 FE 05 20    244            IF A=5 THEN  JR FALL
A279 02 18 18
A27C E1          245            POP     HL
A27D C3 53 A1    246            JP      KEY
A280             247
A280             248 MOVESTONE
A280 CD 1E 20    249            CALL    #LOC
A283 3E 40       250            LD      A,"@"
A285 CD F4 1F    251            CALL    #PRINT
A288             252 IDOU
A288 E1          253            POP     HL
A289 7D 32 A8    254            LD A,L:LD (XZAHYO),A
A28C A1
A28D 7C 32 A9    255            LD A,H:LD (YZAHYO),A
A290 A1
A291 C3 53 A1    256            JP      KEY
A294             257
A294             258 FALL
A294 CD 1E 20    259            CALL    #LOC
A297 CD F1 1F    260            CALL    #PRINTS
A29A 18 EC       261            JR      IDOU
A29C             262
A29C             263
A29C             264
A29C             265 CLEAR
A29C 3A DA A0    266            LD      A,(ROUND)
A29F FE 14 20    267            IF A=20 THEN  JR ALLCLEAR
A2A2 02 18 07
A2A5 3C          268            INC     A
A2A6 32 DA A0    269            LD      (ROUND),A
A2A9 C3 38 A0    270            JP      NAKAMI
```

```
A2AC             271 ALLCLEAR
A2AC 3E 0C       272            LD      A,$0C
A2AE CD F4 1F    273            CALL    #PRINT
A2B1 CD E2 1F    274            CALL    #MPRINT
A2B4 20 2A 2A    275            DM " *******************" DB $0D
A2B7 2A 2A 2A
A2BA 2A 2A 2A
A2BD 2A 2A 2A
A2C0 2A 2A 2A
A2C3 2A 2A 2A
A2C6 2A 2A 0D
A2C9 20 2A 20    276            DM " *                 *" DB $0D
A2CC 20 20 20
A2CF 20 20 20
A2D2 20 20 20
A2D5 20 20 20
A2D8 20 20 20
A2DB 20 2A 0D
A2DE 20 2A 20    277            DM " * CONGRATULATIONS *" DB $0D
A2E1 43 4F 4E
A2E4 47 52 41
A2E7 54 55 4C
A2EA 41 54 49
A2ED 4F 4E 53
A2F0 20 2A 0D
A2F3 20 2A 20    278            DM " *                 *" DB $0D
A2F6 20 20 20
A2F9 20 20 20
A2FC 20 20 20
A2FF 20 20 20
A302 20 20 20
A305 20 2A 0D
A308 20 2A 2A    279            DM " *******************" DB $0D,00
A30B 2A 2A 2A
A30E 2A 2A 2A
A311 2A 2A 2A
A314 2A 2A 2A
A317 2A 2A 2A
A31A 2A 2A 0D
A31D 00
A31E C9          280            RET
A31F             281
A31F             282
```

リスト3 碁石拾いデータ部

```
283 ROUNDDATA
284          DM "                "
285          DM "                "
286          DM "                "
287          DM "        ****    "
288          DM "  >**********   "
289          DM "        ****    "
290          DM "                "
291          DM "                "
292          DM "                "
293          DM "                "
294
295          DM "                "
296          DM "                "
297          DM "       *   *    "
298          DM "      * *       "
299          DM "     *   *      "
300          DM "    *     *     "
301          DM "   *********    "
302          DM " >  *     *     "
303          DM "                "
304          DM "                "
305
306          DM "****************"
307          DM "****************"
308          DM " @@@@ @@@@@ @@@@"
309          DM "  @@@@ @@@* *@@@"
310          DM "   @@@@ @* *@@@@"
311          DM "  @ @@@@O *@@@@>"
312          DM " @@@ @@@@ @@@@  "
313          DM "@@@@@ @@@@@@@   "
314          DM "****************"
315          DM "****************"
316
317          DM "                "
318          DM "                "
319          DM " >  ****        "
320          DM "     ****       "
321          DM "      ****      "
322          DM "       ****     "
323          DM "        ****    "
324          DM "                "
325          DM "                "
326          DM "                "
327
328          DM "                "
329          DM "                "
330          DM " ■@■■>■■ ■■■■ ■ "
331          DM " ■  ■  ■    *■* "
332          DM " ■  ■  ■   *■*  "
333          DM " ■  ■  ■  *■*   "
334          DM " ■  ■  ■ *■*    "
335          DM " ■@ ■@ ■ ■ ■■■■ "
336          DM "                "
337          DM "                "
338
339          DM "                "
340          DM ">      ***      "
341          DM "         *  *   "
342          DM "         *  *   "
343          DM "      *******   "
344          DM "      *  *      "
345          DM "      *  *      "
346          DM "         ***    "
347          DM "                "
348          DM "                "
349
350
351          DM "O              O"
352          DM "    O**O**O     "
353          DM "    *  *  *     "
354          DM "    *  *  *     "
355          DM "    O**O**O     "
356          DM "    *>@*  *     "
```

```
357          DM "    *  *  *     "
358          DM "    O**O**O     "
359          DM "                "
360          DM "O              O"
361
362          DM "                "
363          DM "         *      "
364          DM "       *****    "
365          DM "      *******   "
366          DM ">    *********  "
367          DM "     *********  "
368          DM "     ■■■■■■■■■  "
369          DM "      ■■■■■■■   "
370          DM "       ■■■■■    "
371          DM "        ■■■     "
372
373          DM "                "
374          DM ">     *         "
375          DM "      *         "
376          DM "    ******      "
377          DM "      *  *      "
378          DM "      *  *      "
379          DM "      *  *      "
380          DM "      *  * *    "
381          DM "     **  ***    "
382          DM "                "
383
384          DM ">               "
385          DM "                "
386          DM " @ @ @ @ @ @ @ @"
387          DM "@ @ @ @ @ @ @ @ "
388          DM "OOOOOOOOOOOOOOOO"
389          DM "OOOOOOOOOOOOOOOO"
390          DM "OOOOOOOOOOOOOOOO"
391          DM "OOOOOOOOOOOOOOOO"
392          DM "OOOOOOOOOOOOOOOO"
393          DM "****************"
394
395          DM "OOOOOOOOOOOOOOOO"
396          DM "O>@            O"
397          DM "O              O"
398          DM "O        *     O"
399          DM "O       * **   O"
400          DM "O       * *    O"
401          DM "O      * * *   O"
402          DM "O      * * *   O"
403          DM "O       * *    O"
404          DM "OOOOOOOOOOOOOOOO"
405
406          DM "OOOOOOOOOOOOOOOO"
407          DM "O              O"
408          DM "O   @@@@@@@    O"
409          DM "O   @     @    O"
410          DM "O   @ *O> @    O"
411          DM "O   @     @    O"
412          DM "O   @@@@@@@    O"
413          DM "O              O"
414          DM "O              O"
415          DM "OOOOOOOOOOOOOOOO"
416
417          DM "O@O@O@O@O@O@O@O@"
418          DM "@O@O@O@O@O@O@O@O"
419          DM "O@O@O@O@O@O@O*O@"
420          DM "@O@O@O*O@O@O@O@O"
421          DM "O@O*O@O@O@O@O@O@"
422          DM "@O@O@O@O*O@O@O@O"
423          DM "O*O@O@O@O@O@O@O@"
424          DM "@O@O@O@O@O@O*O@O"
425          DM "O@O@O@O*O@O@O@O@"
426          DM "@O@O@O@O@O@O@O@>"
427
428          DM "@ @*@ @*@ @*@ @*"
429          DM "@ @ @ @ @ @ @ @ "
430          DM "@ @ @ @ @ @ @ @ "
```

```
431          DM "> @             "
432          DM " @ @ @ @ @ @ @ @"
433          DM " @ @ @ @ @ @ @ @"
434          DM " @ @ @ @ @ @ @ @"
435          DM " @ @ @ @ @ @ @ @"
436          DM " @ @ @ @ @ @ @ @"
437          DM " @ @ @ @ @ @ @ @"
438
439          DM "OOOOOOOOOOOOOOOO"
440          DM "O>@            O"
441          DM "O              O"
442          DM "O     *****    O"
443          DM "O      *  *    O"
444          DM "O      *  *    O"
445          DM "O      ****    O"
446          DM "O         *    O"
447          DM "O              O"
448          DM "OOOOOOOOOOOOOOOO"
449
450          DM "    ■  *        "
451          DM " ■      O O ■*O "
452          DM "■ *         ■   "
453          DM " *            ■ "
454          DM "  >   * O *    ■"
455          DM "   O O       O  "
456          DM "O     * * *     "
457          DM " *     *  O■■   "
458          DM "   ■     *      "
459          DM "     O  ■■    ■ "
460
461          DM "  @  @       *  "
462          DM "*■■■■           "
463          DM "  ■■   ■■■■■■■  "
464          DM "  ■■   ■    *   "
465          DM "  ■■  *■ @      "
466          DM "  ■■   ■■■■■■■  "
467          DM "  ■■   ■■■      "
468          DM " ■■■■  ■■■ @  > "
469          DM "O      ■■■      "
470          DM "  *    *     @ @"
471
472          DM "O              O"
473          DM "                "
474          DM "  O***OO******  "
475          DM "  O*O****O**O*  "
476          DM "  O******O****  "
477          DM "  *****O****OO  "
478          DM "  ***********O  "
479          DM "  >**OO**O***O  "
480          DM "                "
481          DM "O              O"
482
483          DM " ■              "
484          DM "*               "
485          DM " ■ @  OOOOO  @  "
486          DM "■■  @ OOOOO @   "
487          DM "     @OO*OO@    "
488          DM "    @ OOOOO @   "
489          DM "   @  OOOOO  @  "
490          DM "                "
491          DM "                "
492          DM "               >"
493
494          DM "   O■ O   O   O "
495          DM "        O ■ * * "
496          DM "■   O @*   O   @"
497          DM " @O* *  @ @*@OO "
498          DM " O @* > ***     "
499          DM "O  * @      O @O"
500          DM " *@O@O■ @ @  ** "
501          DM "  O    O  *   ■ "
502          DM " O  @*   @*O    "
503          DM "     O  O    O @"
```

S-OS"SWORD", グラフィックパッケージMAGIC, Fuzzy BASIC要

漢字出力パッケージ

JACKWRITE

MAGICの用途はゲームやアニメーションだけではありません。今回はS-OS上での漢字表示に挑戦してみましょう。漢字ROMはいりません。データはディスク上から直接読み込むのです。MAGICの可能性を示すユニークなパッケージですね。

Mori Kiichiro
森　喜一郎

S-OSで漢字を

すでにMZ-2500版"SWORD"ではふつうのS-OSのキャラクタセットとともに漢字を使用することも可能となっていますが，漢字VRAMを持たないそのほかの機種では漢字表示などは到底できそうにない状況にあります。そもそもS-OSには漢字を使うという発想自体がそぐわないのでしょうか？　しかしS-OSのキャラクタセットを見ると，シフトJISコードで使用される部分はちゃんとリザーブされているようですし，S-OSは成長するシステムなのですからいずれは漢字だって扱えるようになるのでしょう。

現時点ではグラフィックパッケージMAGICとS-OSの世界が融合して，ようやくS-OSでグラフィックを扱うことが可能になった段階ですから，漢字表示法としてはMAGICを使用したものがもっとも妥当と思われます。そこでMAGICを使用してグラフィック画面に漢字を表示しようじゃないかというのがこのパッケージの発想点なのです。

原理

このパッケージは文字データを受け取ると，表示すべき漢字のJISコードに応じてデータディスクからストロークデータを読み出し，設定されている位置，大きさに変換し描画します。

ふつう漢字表示というのはROM上のフォントデータをG-RAM上に展開したり，漢字VRAMに漢字コードを送ることによってハードウェア的に表示を行うのが一般的です。しかしこのプログラムでは漢字データをストローク（ベクトル）データとしてディスク上に登録し，表示の際そのデータを読み込んでMAGICに渡し描画するという方法をとっています。点(ドット)ではなく線(ベクトル)で漢字を表示するのです。

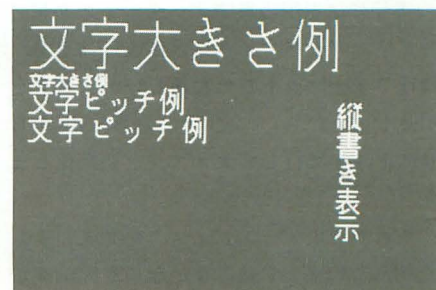
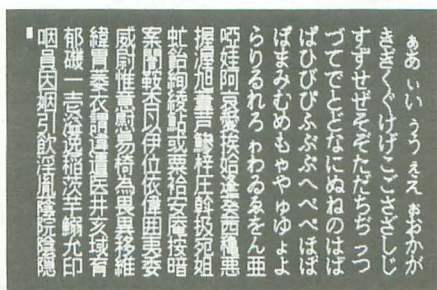
この方式はフォントを使う方法が16×16ドットの場合1文字に32バイトを使用するのに対して，2倍の64バイトを消費するなどデータが大きくなったり，ディスクをアクセスするので表示速度が遅くなるなどの欠点を持っていますが，それなりに長所もあるのです。まず文字の大きさや表示位置を容易に変更できる。ストローク方式の場合は拡大表示などをした場合でも美しい文字が表示できるなどです。さらにS-OSには漢字ROMの内容を読み出すルーチンなどはありませんから，各機種用の漢字ROMを使用するといったことはできません。その代わりMAGICという非常に高速なグラフィックパッケージが用意されていますからデータさえ用意すれば簡単に全機種共通の漢字出力ルーチンができあがるのです(MAGICとディスクドライブは必要ですが)。

さらにMAGICを使用することによって，プログラム側でグラフィックの解像度を気にしなくても全機種で同様な画面表示を得ることができるという利点も発生します。というのも，これまではMZ-1500では文字表示は40字に制限され，X1では40字モードではMAGICが使用できないためグラフィックと文字に別々の表示モードを使用するということが行われていたのですが，文字をグラフィック画面に書けばこのような問題もなくなりました。

ストロークデータの構造は1文字あたり一律に64バイトの構成になっており，内訳は，

　線分の総数（1バイト）
　線分始点座標
　(上位4ビット：Y，下位4ビット：X)
　線分終点座標
　(上位4ビット：Y，下位4ビット：X)
　　⋮
　　⋮

となっています。ひとつの漢字は最大31ストロークで構成され，各ストロークは16×16の分解能で設定されています。つまりこの方式だと32角以上の漢字は出力できないことになりますが，31ストロークで書けない漢字はJIS第1水準にはおそらくないでしょう。ちなみにJIS第2水準の「鬱」という字も31ストローク以内で書けます。

これらのデータはディスクの$20～$49Fレコードを占有し，1レコードあたり4

文字，１枚のディスクでJIS第１水準の漢字/非漢字をすべて格納します。

JISコードからディスク上のデータ位置を求めるには次の式を用いてください。

C＝漢字コード－&H2121
N＝(C￥256) ＊94＋C　MOD256
レコード番号＝N￥4＋32
レコード上の位置＝(N MOD 4)＊64

使用法

このパッケージは表１に示されるような15個の機能を持ったルーチンから成り立っています（ただし14番目と15番目はリザーブされています)。

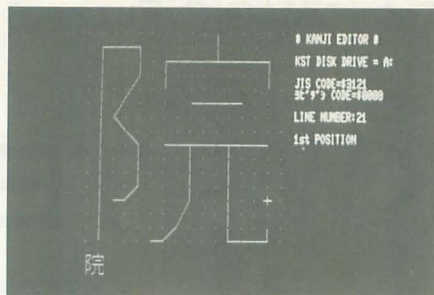
まず，先頭アドレスAB00Hをコールすると漢字の表示をするために必要なMAGICや表示の初期化が行われます。この段階で表示位置や書き込みプレーン，ウィンドウ，表示色，文字の大きさ，文字間隔，縦書き/横書きの指定値にデフォルト値が与えられるのです。このパッケージを使用するときには必ずこのルーチンを呼び出すようにしてください。

表示ルーチンは１文字表示，メッセージ表示，文字列表示などS-OSの文字表示とほぼ同様なものが揃えてあります。そのほかのルーチンは表示の制御，漢字表示時のパラメータを設定するルーチン群です。具体的な使用法は表１およびサンプルプログラムを参考にしてください。

リスト３は画面にひらがなを表示するサンプルプログラムです。デフォルトドライブにデータディスクをセットしてJ8000で起動します。

プログラム/データの入力

リスト１は漢字出力ルーチンの本体です。MACINTO-Cなどのマシン語入力ツールから打ち込んでください。このプログラムの実行には専用の漢字データディスクが必要です。データディスクの作成にはまず，物理フォーマットされた新しいディスクを用意してください。次にFuzzyBASICからリスト４を実行してディスクを初期化します。

これで専用データディスクが完成しましたが，このままでは，まだなにもデータが

入っていませんのでとりあえずサンプルデータとしてひらがなのデータをディスクに転送しましょう。リスト９を入力してください。Aドライブにデータディスクを入力J6000で起動すると，データディスクにメモリ上のデータを書き込みます。漢字データは通常のファイルコピーではバックアップできません。データディスクのバックアップにはリスト２のユーティリティを使用してください。ただしリスト２は80トラック専用となっていますので70トラックのディスクを使用されている方は36A7Hを46Hに修正してください。このツールはあくまでもデータディスク専用のものですので通常のディスクバックアップの際の動作は保証されません。また，データディスクにはそのほかのファイルを入れないようにしてください。

サンプルデータはひらがなだけですので，このままではひらがなしか表示できません。必要なデータは各自で拡張していくしかないのです。新たにデータを作成する場合はリスト３のFuzzyBASIC版のエディタプログラムを使用します。作成したい漢字のコードと呼び出したいデータを指定してエディットします。データを呼び出す必要のないときはそのままリターンキーだけを押してください。漢字ROMのフォントを参考にすれば，意外と簡単にデータを作成することができます（MZ-1500では変数WIDの値を40としてください)。

X1やMZ-2500などではちょっと工夫してリスト５のプログラムをそれぞれのBASIC用に移植し，さらに漢字ROMから読み出したデータを拡大してそれを下敷きにするという方法を用いれば，あっというまにデータを作ることができます。

最後に

さて，このままではこのルーチンはMAGICを使ったアプリケーションのメッセージを日本語化するという程度にしか活用できません。現状では日本語入力フロントプロセッサもありませんし，画面上の漢字をプリンタに出力する方法もありません。共通バスにRS-232Cが載ったことですし，将来全機種共通のプロッタドライバでもできればいろいろと面白いことができそうですね。

今後S-OSで漢字ROMを使用し，ドットデータで漢字表示をするようになったとしても，ストロークデータが用意されているということは非常に意義深いものとなるでしょう。とにかくS-OSの可能性がまたひとつ広がったわけです。皆さんも行書体や丸文字など，自分専用の漢字データを作ったり，このパッケージを使ったアプリケーションを考えてみてください。

Profile
◇森さんは大阪府にお住まいの21歳，昼間は公務員，夜は大学生という毎日です。現在X1turbo，MZ-80Bユーザー，大阪工大S-OSユーザーズクラブのメンバーでもあります。

表１　内部ルーチン

アドレス	名称	機能
AB00H	KPINIT	MAGIC，表示関係を初期化する
AB03H	KPRINT	１文字表示，DEレジスタの値をJISコードとみなし表示する
AB06H	KMPRNT	直後のアドレスからの文字列を表示する。エンドコードは0000H
AB09H	KSPRNT	DEで示されるアドレスの文字列を表示する。エンドコードは0000H
AB0CH	KLTNL	改行する
AB0FH	KPCLS	書き込み中の画面をクリアする
AB12H	STSCRN	SCREEN設定。Aレジスタのプレーンに書き込む
AB15H	SETWIN	WINDOW設定。直後のデータ列をMAGICに渡す
AB18H	SETPAL	PALET設定。直後のデータ列をMAGICに渡す
AB1BH	STSIZE	DレジスタにY，EレジスタにXのドット数を指定
AB1EH	STPICH	行間を指定する。Aレジスタの上位４ビットにY，下位４ビットにXの倍率(n/16)を指定する。STSIZEと連動する
AB21H	SETDIR	縦書き/横書きを指定する。A＝0：横，A＝0：縦
AB24H	SETLOC	直後のアドレスの座標から書き始めるよう指定する（Ex. DW　X，Y）
AB27H	────	リザーブ
AB2AH	────	リザーブ

リスト1　漢字表示ルーチン

```
AB00 C3 2D AB C3 8D AB C3 B2 : 0B
AB08 AB C3 BA AB C3 C1 AB C3 : C5
AB10 E3 AB C3 1C AC C3 50 AC : D8
AB18 C3 31 AC C3 6F AC C3 74 : B5
AB20 AC C3 78 AC C3 7C AC C3 : 41
AB28 90 AC C3 90 AC C5 D5 E5 : BA
AB30 DD E5 CD 24 20 32 94 AE : 47
AB38 3E 04 11 C8 AE CD A3 1F : 58
AB40 CD 09 20 38 39 21 00 00 : 88
AB48 22 DE AE 22 E0 AE 21 20 : 9F
AB50 10 22 DA AE AF 32 DC AE : 25
AB58 32 DD AE 2A E4 AE 11 0B : 95
AB60 00 ED 52 22 97 AE CD 00 : 73
AB68 AF 21 B2 AE 11 9C AE 01 : 8C
AB70 16 00 ED B0 DD 21 9C AE : FB
AB78 CD 04 B0 B7 18 09 F5 AF : FD
---------------------------------
SUM: 2E 1C E4 DE F1 3E 53 41 164B

AB80 32 94 AE CD C4 1F F1 DD : F2
AB88 E1 E1 D1 C1 C9 C5 D5 E5 : 9C
AB90 DD E5 CD 16 AE 38 15 CD : 6D
AB98 1D AE 38 10 32 5D 1F 2A : EB
ABA0 E2 AE 22 95 AE CD D3 AC : 41
ABA8 CD 50 AE B7 DD E1 E1 D1 : F2
ABB0 C1 C9 DD E1 CD 91 AC DD : 2F
ABB8 E5 C9 D5 DD E1 CD 91 AC : 4B
ABC0 C9 D5 E5 3A DD AE E6 01 : 2F
ABC8 20 0C 11 80 02 ED 53 DE : DD
ABD0 AE CD 14 AD 18 0A 11 C8 : 37
ABD8 00 ED 53 E0 AE CD 14 AD : 5C
ABE0 E1 D1 C9 C5 D5 E5 DD E5 : BC
ABE8 DD 21 1A AC CD 04 B0 2A : 6F
ABF0 9F AE 22 E0 AE 3A DD AE : C2
ABF8 E6 01 20 08 2A 9D AE 22 : A6
---------------------------------
SUM: 3C D4 88 5E C5 B7 61 F2 DF4A

AC00 DE AE 18 10 2A A1 AE 3A : 67
AC08 DA AE 3D 06 00 4F B7 ED : BE
AC10 42 22 DE AE DD E1 E1 D1 : 60
AC18 C1 C9 09 0F C5 D5 E5 DD : FE
AC20 E5 DD 21 9C AE DD 77 14 : 95
AC28 CD 04 B0 DD E1 E1 D1 C1 : B2
AC30 C9 E1 C5 D5 DD E5 11 A6 : BD
AC38 AE 06 08 7E 23 12 13 10 : 92
AC40 FA E5 DD 21 9C AE CD 04 : F8
AC48 B0 E1 DD E1 D1 C1 E5 C9 : 8F
AC50 E1 C5 D5 DD E5 11 9D AE : 99
AC58 06 08 7E 23 12 13 10 FA : DE

AC60 E5 DD 21 9C AE CD 04 B0 : AE
AC68 E1 DD E1 D1 C1 E5 C9 ED : CC
AC70 53 DA AE C9 32 DC AE C9 : 29
AC78 32 DD AE C9 E1 D5 5E 23 : BD
---------------------------------
SUM: C0 13 45 A0 41 51 CF 5E BE18

AC80 56 23 ED 53 DE AE 5E 23 : C6
AC88 56 23 ED 53 E0 AE D1 E5 : FD
AC90 C9 C5 D5 E5 CD 16 AE 38 : 11
AC98 21 2A E2 AE 22 95 AE 32 : 72
ACA0 5D 1F CD 1D AE 38 13 DD : 3C
ACA8 56 01 DD 5E 00 DD 23 DD : 6F
ACB0 23 7A B3 28 16 CD D3 AC : DA
ACB8 18 E8 DD 56 01 DD 5E 00 : 6F
ACC0 DD 23 DD 23 7A B3 20 F2 : 3F
ACC8 37 18 04 CD 50 AE B7 E1 : B6
ACD0 D1 C1 C9 CD C4 AD 38 3C : 0D
ACD8 2A 64 1F 3E 01 CD 00 20 : D9
ACE0 09 DD E5 E5 DD E1 DD 7E : C9
ACE8 00 DD 23 B7 28 24 2A 95 : C2
ACF0 AE 47 37 DD 4E 01 CD FA : 1F
ACF8 AD C4 8A AD DD 4E 00 79 : 4C
---------------------------------
SUM: F7 DC 5D 53 31 F5 D5 8D A792

AD00 32 99 AE CD 8A AD DD 23 : 7D
AD08 DD 23 22 95 AE CD 46 AE : 26
AD10 10 E1 DD E1 3A DD AE E6 : 5A
AD18 01 20 12 B7 CD 3F AD CD : 70
AD20 1D AE D0 21 00 00 22 DE : BC
AD28 AE CD 67 AD C9 CD 67 AD : 39
AD30 CD 1D AE D0 21 00 00 22 : AB
AD38 E0 AE 37 CD 3F AD C9 F5 : 3C
AD40 2A DE AE 3A DA AE 16 00 : 8E
AD48 5F 3A DC AE E6 0F CD 7A : 5F
AD50 AE CD 69 AE 3A DA AE CD : 21
AD58 8F AE F1 38 03 19 18 03 : 9D
AD60 B7 ED 52 22 DE AE C9 2A : 97
AD68 E0 AE 3A DB AE 16 00 5F : C6
AD70 3A DC AE E6 F0 0F 0F 0F : C7
AD78 0F CD 7A AE CD 69 AE 3A : 22
---------------------------------
SUM: 3E DA 73 C4 AE FC FF 42 B759

AD80 DB AE CD 8F AE 19 22 E0 : AE
AD88 AE C9 16 00 59 CB 3B CB : B7
AD90 3B CB 3B CB 3B 3A DA AE : 09
AD98 CD 7A AE CD 69 AE E5 2A : E8
ADA0 DE AE 19 EB E1 73 23 72 : 79

ADA8 23 79 E6 0F 16 00 5F 3A : 40
ADB0 DB AE CD 7A AE CD 69 AE : 62
ADB8 E5 2A E0 AE 19 EB E1 73 : F5
ADC0 23 72 23 C9 EB 11 21 21 : BF
ADC8 B7 ED 52 3E 5D 95 D8 3E : 3C
ADD0 2E 94 D8 16 00 5C 3E 5E : A8
ADD8 CD 7A AE 7D CD 8F AE 7B : F7
ADE0 E6 03 87 87 87 87 87 87 : 13
ADE8 06 00 4F CB 3A CB 1B CB : 0B
ADF0 3A CB 1B 3E 20 CD 8F AE : 88
ADF8 B7 C9 38 0E 3A 99 AE A9 : F0
---------------------------------
SUM: 04 BF 9C 81 99 40 AC 31 718A

AE00 20 08 E5 2A 9A AE 34 E1 : 94
AE08 AF C9 AF 77 23 3E 02 77 : 78
AE10 22 9A AE 23 B7 C9 3A 94 : DB
AE18 AE B7 C0 37 C9 ED 4B DE : 3B
AE20 AE 2A A1 AE B7 ED 42 D8 : E5
AE28 2A 9D AE B7 ED 42 28 02 : 85
AE30 3F D8 ED 4B E0 AE 2A A3 : AA
AE38 AE B7 ED 42 D8 2A 9F AE : E3
AE40 B7 ED 42 C8 3F C9 ED 5B : FE
AE48 97 AE EB B7 ED 52 EB D0 : E1
AE50 2A 95 AE 36 0F 2A E2 AE : 6C
AE58 22 95 AE E5 DD E5 E5 DD : CE
AE60 E1 CD 04 B0 DD E1 E1 37 : 38
AE68 C9 CB 3A CB 1B CB 3A CB : 84
AE70 1B CB 3A CB 1B CB 3A CB : D6
AE78 1B C9 E5 21 00 00 B7 28 : C9
---------------------------------
SUM: DE 69 11 EE C4 4A 99 A0 3499

AE80 0B CB 3F 30 01 19 CB 23 : 4D
AE88 CB 12 18 F2 EB E1 C9 83 : FF
AE90 5F D0 14 C9 00 00 00 00 : 0C
AE98 00 00 00 00 00 00 00 00 : 00
AEA0 00 00 00 00 00 00 00 00 : 00
AEA8 00 00 00 00 00 00 00 00 : 00
AEB0 00 00 06 00 00 00 00 7F : 85
AEB8 02 C7 00 0A 00 07 00 07 : E1
AEC0 00 07 00 07 07 02 00 0F : 26
AEC8 4B 53 54 20 20 20 20 20 : 92
AED0 20 20 20 20 20 2E 44 49 : 5B
AED8 43 00 20 10 00 00 00 00 : 73
AEE0 00 00 00 A0 FF AA       : 49
---------------------------------
SUM: E5 EE 05 EC 32 FB F8 A4 0E23
```

リスト2　ディスクバックアップ

```
3000 18 1B 1A B7 28 17 CD E3 : F3
3008 30 32 1E 31 CD 07 31 C2 : 78
3010 18 31 CD E3 30 32 1F 31 : AB
3018 CD 07 31 18 54 CD E2 1F : 3F
3020 53 4F 55 52 43 45 20 20 : 11
3028 20 20 20 20 44 45 56 49 : A8
3030 43 45 20 3D 3D 3E 20 00 : 80
3038 CD 21 20 32 1E 31 CD F4 : 50
3040 1F CD EE 1F CD E8 30 CD : AB
3048 E2 1F 44 45 53 54 49 4E : C8
3050 41 54 49 4F 4E 20 44 45 : 24
3058 56 49 43 45 20 3D 3D 3E : FF
3060 20 00 CD 21 20 32 1F 31 : B0
3068 CD F4 1F CD EE 1F CD E8 : 6F
3070 30 3A 1E 31 CD F4 1F CD : 66
3078 E2 1F 3A 20 3D 3D 3E 20 : 33
---------------------------------
SUM: 47 30 ED FB 01 31 A5 F6 C7EC

3080 00 3A 1F 31 CD F4 1F CD : 37
3088 E2 1F 3A 20 20 53 75 72 : B5
3090 65 28 59 2F 4E 29 3F 20 : EB
3098 00 CD 21 20 CD F4 1F CD : BB
30A0 EE 1F D6 59 B7 C0 3E 50 : 41
30A8 01 10 00 11 00 00 21 20 : 63
30B0 31 F5 3A 1E 31 32 5D 1F : 5D
30B8 79 CD 00 20 38 5E 3A 1F : 55
30C0 31 32 5D 1F 79 CD 03 20 : 48
30C8 38 52 EB 09 EB F1 3D 20 : B7
30D0 E0 CD E2 1F 43 6F 6D 70 : 3D
30D8 6C 65 74 65 64 20 21 0D : 5C
30E0 00 B7 C9 CD 10 31 1A 13 : BB
30E8 FE 41 C8 FE 42 C8 FE 43 : 50
30F0 C8 FE 44 C8 FE 45 20 0B : 40
30F8 F5 3A FF 20 FE 50 20 02 : BE
---------------------------------
SUM: 50 25 55 A7 81 8F 0E FA C57A

3100 F1 C9 F1 D1 C3 18 31 1A : A2
3108 13 FE 3A C8 D1 C3 18 31 : F0
3110 1A 13 FE 20 28 FA 1B C9 : 51
3118 3E 03 37 C9 C1 C9 00 00 : CB
---------------------------------
SUM: 5C DD 60 82 7D 9E 64 14 8A50
```

リスト3　サンプルプログラム

```
8000 CD 00 AB D8 CD 15 AB 00 : DD
8008 00 00 00 7F 02 BF 00 11 : 51
8010 20 10 CD 1B AB 3E 01 CD : CF
8018 21 AB 3E 05 CD 1E AB CD : 72
8020 0F AB DD 21 40 80 DD 56 : AB
8028 01 DD 5E 00 7A B3 C8 DD : 0E
8030 46 02 CD 03 AB 13 10 FA : E0
8038 DD 23 DD 23 DD 23 18 E6 : FE
8040 21 24 53 21 30 5E 21 31 : 99
8048 03 00 00                : 03
---------------------------------
SUM: 65 8C EE DF B9 F7 45 EF 5762
```

リスト4　データディスク初期化

```
100 ' create KST file      1987/03/13  by K.MORI
110 '
120 limit &HADFF' AE00-AEFF = RECORD BUFFER
130 '
140 「SETDISK」
150  repeat
160    print "FORMATTED NEW DISK DRIVE ":D=flash:print chr$(D);":"/
170    print "Ready(Y/N)?":A=flash:print chr$(A)/
180  until A=89
190 '
200 「DIRINIT」
210  devi D,&HAE00,16,1
220  if peek(&HAE00)<>255 then beep :print "THIS IS USED DISK."/:end
230  poke &HAE00,4:mem &HAE01,"KST          DIC "
240  poke &HAE1D,0,2,0
250  devo D,&HAE00,16,1
260 '
270 「FATINIT」
280  devi D,&HAE00,14,1
290  for I=2 to &H46
300    poke &HAE00+I,I+1
310  next :poke &HAE47,&H8F
320  devo D,&HAE00,14,1
330 '
340 「DATAINIT」
350  for I=0 to 127
```

```
360    wpoke &HAE00+I*2,0
370  next
380  for I=32 to 1135
390    print "RECORD ";I:print
400    devo D,&HAE00,I,1
410  next
420  '
430 print "COMPLETED."/:end
```

●BASICチェックサム

100:96	110:27	120:C3	130:27	140:CF	150:84	160:82	170:40
180:74	190:27	200:CB	210:7E	220:CE	230:32	240:24	250:7F
260:27	270:C7	280:7C	290:91	300:3E	310:D9	320:7D	330:27
340:06	350:51	360:26	370:83	380:B6	390:0D	400:61	410:83
420:27	430:A8						

リスト5　漢字データエディタ

```
10 ' KANJI INPUT EDITOR   1987/05/03  by  K.MORI
20 '
30 limit &H9FFF:graph 1:window (0,0)-(639,199)
40 color 1,1:palet 0,1,2,3,4,5,6,7:cls:WID=80:width WID
50 print tab(WID/2);"* KANJI EDITOR *"//
60 print tab(WID/2);"KST DISK DRIVE = "
70 D=flash:if D<&H41 or D>&H45 goto 70
80 print chr$(D);":"/
90 '
100 repeat
110   locate WID/2,4:print "JIS": gosub 「INPUT」
120   if C=0 then cls2:end: else cls1
130   gosub 「DOTSET」:X=0:Y=0
140   gosub 「READ」
150   gosub 「GCUR」
160   locate WID/2,7:print "LINE NUMBER:";L/2;"   "
170   locate WID/2,9:print "1st POSITION"
180   repeat:until get=0
190   '
200   repeat
210     A=inkey
220     if A="1" then V=-1:W=1: gosub 「MOVE」
230     if A="2" then V=0:W=1: gosub 「MOVE」
240     if A="3" then V=1:W=1: gosub 「MOVE」
250     if A="4" then V=-1:W=0: gosub 「MOVE」
260     if A="5" then  gosub 「LINE」
270     if A="6" then V=1:W=0: gosub 「MOVE」
280     if A="7" then V=-1:W=-1: gosub 「MOVE」
290     if A="8" then V=0:W=-1: gosub 「MOVE」
300     if A="9" then V=1:W=-1: gosub 「MOVE」
310     if A="0" then  gosub 「DELETE」
320     if A=13 then  gosub 「WRITE」
330     repeat:until get=0
340     locate WID/2+12,7:print L/2
350   until A=13 or A=27
360 until A=27:cls2:end
370 '
380 ' SUBROUTINE
390 '
400 「INPUT」
410  print " CODE=$":X=curx-1:Y=cury
420  locate X,Y:input "";C:if C=0 goto 450
430  if C/256>&H4F or mod(C,256)>&H7E goto 420
440  if C/256<&H21 or mod(C,256)<&H21 goto 420
450  X=0:Y=0
460 return
470 '
480 「DOTSET」
490  cls1
500  for X=0 to 15
510    for Y=0 to 15:dot (X*20,Y*10):next
520  next
530 return
540 '
550 「MOVE」
560  gosub 「GCUR」:X=X+V:Y=Y+W
570  if X=16 then X=0: else if X=65535 then X=15
580  if Y=16 then Y=0: else if Y=65535 then Y=15
590   gosub 「GCUR」
600 return
610 '
620 「GCUR」
630 if mod(L,2)
640  then
650    line (R*20,S*10)-(X*20,Y*10)
660  else
670    line (X*20-6,Y*10)-(X*20+6,Y*10)
680    line (X*20,Y*10-3)-(X*20,Y*10+3)
690  end if
700  dot (X*2,Y+160)
710 return
720 '
730 「LINE」
740  if L=62 then beep : return
750  locate WID/2,9:print "2nd": gosub 「GCUR」
760  if mod(L,2)
770    then line (R*20,S*10)-(X*20,Y*10)
780    color 3,1:line (R*2,S+160)-(X*2,Y+160):color 1,1
790    wpoke &HA000+B+L,rotld(rotld(rotld(R)+S)+X)+Y
800    locate WID/2,9:print "1st"
810  end if
820  R=X:S=Y:L=L+1: gosub 「GCUR」
830 return
840 '
850 「DELETE」
860  if L=0 then beep : return
870  gosub 「GCUR」
880  L=L-1:X=R:Y=S
890  locate WID/2,9:print "1st"
900  if mod(L,2)=1
910    then window (0,160)-(31,175)
920    cls1:color 3,1
930    for I=1 to L step 2
940      P=wpeek(&HA000+B+I)
950      Y=P and 15:X=(P/16) and 15
960      S=(P/256) and 15:R=(P/4096) and 15
970      if I<L then line (R*2,S+160)-(X*2,Y+160)
980    next :wpoke &HA000+B+L,0
990    color 1,1:window (0,0)-(639,199)
1000   line (R*20,S*10)-(X*20,Y*10)
1010   locate WID/2,9:print "2nd"
1020 end if
1030  gosub 「GCUR」
1040 return
1050 '
1060 「READ」
1070  N=C-&H2121
1080  N=(N/256)*94+mod(N,256)
1090  B=mod(N,4)*64:N=N/4+32
1100  devi D,&HA000,N,1
1110  L=peek(&HA000+B)
1120  if L=0 then  gosub 「READ2」
1130  if L=0 then  return : else L=L*2-1
1140  for I=1 to L step 2
1150    P=wpeek(&HA000+B+I)
1160    Y=P and 15:X=(P/16) and 15
1170    S=(P/256) and 15:R=(P/4096) and 15
1180    line (R*20,S*10)-(X*20,Y*10):color 3,1
1190    line (R*2,S+160)-(X*2,Y+160):color 1,1
1200  next :L=L+1
1210 return
1220 '
1230 「READ2」
1240  print tab(WID/2);"ﾖﾋﾞﾀﾞｼ"
1250  locate curx+7,cury:print "0"
1260  locate curx-8,cury: gosub 「INPUT」
1270  if C=0 then L=0: return
1280  M=C-&H2121
1290  M=(M/256)*94+mod(M,256)
1300  C=mod(M,4)*64:M=M/4+32
1310  devi D,&HA100,M,1
1320  L=peek(&HA100+C)
1330  if L>0 then ldir &HA100+C,&HA000+B,64
1340 return
1350 '
1360 「WRITE」
1370  poke &HA000+B,L/2
1380  devo D,&HA000,N,1
1390 return
```

●BASICチェックサム

10:75	20:27	30:5B	40:B4	50:4B	60:59	70:F9	80:66
90:27	100:84	110:7A	120:FC	130:13	140:5D	150:72	160:9E
170:FB	180:47	190:27	200:84	210:14	220:B2	230:85	240:87
250:B4	260:7D	270:89	280:E5	290:B8	300:BA	310:03	320:CB
330:47	340:83	350:CB	360:43	370:27	380:57	390:27	400:48
410:B6	420:D5	430:41	440:0D	450:C5	460:8A	470:27	480:8B
490:C8	500:2C	510:3D	520:83	530:8A	540:27	550:EF	560:C5
570:5D	580:61	590:72	600:8A	610:27	620:E9	630:E7	640:8F
650:81	660:90	670:51	680:4B	690:91	700:20	710:8A	720:27
730:E0	740:89	750:FE	760:E7	770:10	780:ED	790:D3	800:66
810:91	820:21	830:8A	840:27	850:6B	860:51	870:72	880:77
890:66	900:55	910:58	920:78	930:B8	940:69	950:B0	960:2D
970:7D	980:E2	990:54	1000:81	1010:52	1020:91	1030:72	1040:8A
1050:27	1060:D4	1070:2F	1080:7E	1090:76	1100:50	1110:F0	1120:65
1130:B9	1140:B8	1150:69	1160:B0	1170:2D	1180:31	1190:3D	1200:EE
1210:8A	1220:27	1230:06	1240:DE	1250:E5	1260:AF	1270:4A	1280:2E
1290:7B	1300:74	1310:50	1320:F2	1330:4E	1340:8A	1350:27	1360:43
1370:2A	1380:51	1390:8A					

リスト6　漢字表示ルーチンソースリスト

```
0000            1 ;********************************
0000            2 ;*  KANJI PRINT PACKAGE VER1.0  *
0000            3 ;*        1987/06/10  by K.MORI  *
0000            4 ;********************************
0000            5 ;
0000            6 MGINIT EQU 0AF00H  ; MAGIC INIT
0000            7 MAGIC  EQU 0B004H  ; MAGIC ADR
0000            8 #RDVSW EQU  2024H  ; DEV.  LOAD
0000            9 #ROPEN EQU  2009H  ; FCB   READ
0000           10 #SCTRD EQU  2000H  ; DISK  READ
0000           11 #DTBUF EQU  1F64H  ; REC.  BUFFER
0000           12 #MEMAX EQU  1F6AH  ; MEMORY MAX
0000           13 #FILE  EQU  1FA3H  ; FILE NAME SET
0000           14 #DSK   EQU  1F5DH  ; DEV.  WORK
0000           15 #BELL  EQU  1FC4H  ; BEEP
0000           16 WRKORG EQU 0A000H  ; MAGIC DATA START
0000           17 WRKEND EQU 0AAFFH  ; MAGIC DATA END
0000           18 ;
0000           19 ; JUMP TABLE ----------------------
0000           20 ;
AB00           21        ORG  0AB00H
AB00           22 ;
```


▶昔発表されたキャリーラボのペイントルーチンや3重和音は，MZユーザーに対する餞別だったんでしょうか。　藤井　崇（17）富山県

```
AB00            23 JMPTBL
AB00 C3 2D AB   24         JP   KPINIT          ; INITAL
AB03 C3 8D AB   25         JP   KPRINT          ; CHR.PRINT
AB06 C3 B2 AB   26         JP   KMPRNT          ; MSG.PRINT
AB09 C3 BA AB   27         JP   KSPRNT          ; STR.PRINT
AB0C C3 C1 AB   28         JP   KLTNL           ; CRLF
AB0F C3 E3 AB   29         JP   KPCLS           ; CLS
AB12 C3 1C AC   30         JP   STSCRN          ; SCREEN
AB15 C3 50 AC   31         JP   SETWIN          ; WINDOW
AB18 C3 31 AC   32         JP   SETPAL          ; PALET
AB1B C3 6F AC   33         JP   STSIZE          ; CHR.SIZE
AB1E C3 74 AC   34         JP   STPICH          ; SPC.PITCH
AB21 C3 78 AC   35         JP   SETDIR          ; DIRECTION
AB24 C3 7C AC   36         JP   SETLOC          ; LOCATE
AB27 C3 90 AC   37         JP   RSVTBL          ; RESERVED
AB2A C3 90 AC   38         JP   RSVTBL          ; RESERVED
AB2D            39 ;
AB2D            40 ; ------------------------------------
AB2D            41 ;
AB2D            42 KPINIT
AB2D C5         43         PUSH BC
AB2E D5         44         PUSH DE
AB2F E5         45         PUSH HL
AB30 DD E5      46         PUSH IX
AB32 CD 24 20   47         CALL #RDVSW          ; Acc <== DEV.
AB35 32 94 AE   48         LD   (KSTDEV),A
AB38 3E 04      49         LD   A,04H           ; A <== 'Asc'
AB3A 11 C8 AE   50         LD   DE,KSTNAM
AB3D CD A3 1F   51         CALL #FILE
AB40 CD 09 20   52         CALL #ROPEN
AB43 38 39      53         JR   C,KPIERR        ; File exist ?
AB45 21 00 00   54         LD   HL,0
AB48 22 DE AE   55         LD   (KPCURX),HL     ; HOME
AB4B 22 E0 AE   56         LD   (KPCURY),HL
AB4E 21 20 10   57         LD   HL,1020H        ; H <= Y,L <= X
AB51 22 DA AE   58         LD   (KPSIZE),HL
AB54 AF         59         XOR  A
AB55 32 DC AE   60         LD   (KPITCH),A      ; SPACE PITCH
AB58 32 DD AE   61         LD   (KPDIR),A       ; DIRECTION
AB5B 2A E4 AE   62         LD   HL,(MADRED)
AB5E 11 0B 00   63         LD   DE,11
AB61 ED 52      64         SBC  HL,DE
AB63 22 97 AE   65         LD   (MAEDWK),HL     ; DATA ADR. - 11
AB66 CD 00 AF   66         CALL MGINIT
AB69 21 B2 AE   67         LD   HL,MGIDAT
AB6C 11 9C AE   68         LD   DE,MGDATA
AB6F 01 16 00   69         LD   BC,22
AB72 ED B0      70         LDIR
AB74 DD 21 9C   71         LD   IX,MGDATA
AB77 AE
AB78 CD 04 B0   72         CALL MAGIC
AB7B B7         73         OR   A               ; RESET CF
AB7C 18 09      74         JR   KPIEND
AB7E            75 KPIERR
AB7E F5         76         PUSH AF
AB7F AF         77         XOR  A
AB80 32 94 AE   78         LD   (KSTDEV),A
AB83 CD C4 1F   79         CALL #BELL
AB86 F1         80         POP  AF
AB87            81 KPIEND
AB87 DD E1      82         POP  IX
AB89 E1         83         POP  HL
AB8A D1         84         POP  DE
AB8B C1         85         POP  BC
AB8C C9         86         RET
AB8D            87 ;
AB8D            88 KPRINT                       ; DE = JIS CODE
AB8D C5         89         PUSH BC
AB8E D5         90         PUSH DE
AB8F E5         91         PUSH HL
AB90 DD E5      92         PUSH IX
AB92 CD 16 AE   93         CALL KSTCHK          ; INIT ?
AB95 38 15      94         JR   C,KPREND        ; KST SET ?
AB97 CD 1D AE   95         CALL CURCHK
AB9A 38 10      96         JR   C,KPREND        ; CUR. Over ?
AB9C 32 5D 1F   97         LD   (#DSK),A        ; DEV. SET
AB9F 2A E2 AE   98         LD   HL,(MAGADR)
ABA2 22 95 AE   99         LD   (MADRWK),HL
ABA5 CD D3 AC  100         CALL JISMAG          ; KST ==> MAGIC
ABA8 CD 50 AE  101         CALL GMAGIC          ; KANJI WRITE
ABAB B7        102         OR   A               ; RCF
ABAC           103 KPREND
ABAC DD E1     104         POP  IX
ABAE E1        105         POP  HL
ABAF D1        106         POP  DE
ABB0 C1        107         POP  BC
ABB1 C9        108         RET
ABB2           109 ;
ABB2           110 KMPRNT                       ; (PC) = "MSG"
ABB2 DD E1     111         POP  IX              ; IX <== PC
ABB4 CD 91 AC  112         CALL KMSSUB
ABB7 DD E5     113         PUSH IX              ; NEXT OP.
ABB9 C9        114         RET
ABBA           115 ;
ABBA           116 KSPRNT                       ; (DE) = "STR"
ABBA D5        117         PUSH DE
ABBB DD E1     118         POP  IX              ; IX <== DE
ABBD CD 91 AC  119         CALL KMSSUB
ABC0 C9        120         RET
ABC1           121 ;
ABC1           122 KLTNL
ABC1 D5        123         PUSH DE
ABC2 E5        124         PUSH HL
ABC3 3A DD AE  125         LD   A,(KPDIR)
ABC6 E6 01     126         AND  1
ABC8 20 0C     127         JR   NZ,VERLF
ABCA           128 HORLF
ABCA 11 80 02  129         LD   DE,640
ABCD ED 53 DE  130         LD   (KPCURX),DE
ABD0 AE
ABD1 CD 14 AD  131         CALL CURINC
ABD4 18 0A     132         JR   KLTNL2
ABD6           133 VERLF
ABD6 11 C8 00  134         LD   DE,200
ABD9 ED 53 E0  135         LD   (KPCURY),DE
ABDC AE
ABDD CD 14 AD  136         CALL CURINC
ABE0           137 KLTNL2
ABE0 E1        138         POP  HL
ABE1 D1        139         POP  DE
ABE2 C9        140         RET
ABE3           141 ;
ABE3           142 KPCLS
ABE3 C5        143         PUSH BC
ABE4 D5        144         PUSH DE
ABE5 E5        145         PUSH HL
ABE6 DD E5     146         PUSH IX
ABE8 DD 21 1A  147         LD   IX,CLSDAT
ABEB AC
ABEC CD 04 B0  148         CALL MAGIC
ABEF 2A 9F AE  149         LD   HL,(MGDATA+3)
ABF2 22 E0 AE  150         LD   (KPCURY),HL
ABF5 3A DD AE  151         LD   A,(KPDIR)
ABF8 E6 01     152         AND  1
ABFA 20 08     153         JR   NZ,VERHME
ABFC           154 HORHME
ABFC 2A 9D AE  155         LD   HL,(MGDATA+1)
ABFF 22 DE AE  156         LD   (KPCURX),HL
AC02 18 10     157         JR   KCLSED
AC04           158 VERHME
AC04 2A A1 AE  159         LD   HL,(MGDATA+5)
AC07 3A DA AE  160         LD   A,(KPSIZE)
AC0A 3D        161         DEC  A
AC0B 06 00     162         LD   B,0
AC0D 4F        163         LD   C,A
AC0E B7        164         OR   A               ; RESET CF
AC0F ED 42     165         SBC  HL,BC
AC11 22 DE AE  166         LD   (KPCURX),HL
AC14           167 KCLSED
AC14 DD E1     168         POP  IX
AC16 E1        169         POP  HL
AC17 D1        170         POP  DE
AC18 C1        171         POP  BC
AC19 C9        172         RET
AC1A           173 CLSDAT
AC1A 09 0F     174         DEFB 09H:0FH         ; CLS,DONE
AC1C           175 ;
AC1C           176 STSCRN                       ; A = PLANE
AC1C C5        177         PUSH BC
AC1D D5        178         PUSH DE
AC1E E5        179         PUSH HL
AC1F DD E5     180         PUSH IX
AC21 DD 21 9C  181         LD   IX,MGDATA
AC24 AE
AC25 DD 77 14  182         LD   (IX+20),A
AC28 CD 04 B0  183         CALL MAGIC
AC2B DD E1     184         POP  IX
AC2D E1        185         POP  HL
AC2E D1        186         POP  DE
AC2F C1        187         POP  BC
AC30 C9        188         RET
AC31           189 ;
AC31           190 SETPAL                       ; (PC) = DATA
AC31 E1        191         POP  HL
AC32 C5        192         PUSH BC
AC33 D5        193         PUSH DE
AC34 DD E5     194         PUSH IX
AC36 11 A6 AE  195         LD   DE,MGDATA+10
AC39 06 08     196         LD   B,8
AC3B           197 STPAL2
AC3B 7E        198         LD   A,(HL)
AC3C 23        199         INC  HL
AC3D 12        200         LD   (DE),A
AC3E 13        201         INC  DE
AC3F 10 FA     202         DJNZ STPAL2
AC41 E5        203         PUSH HL
AC42 DD 21 9C  204         LD   IX,MGDATA
AC45 AE
AC46 CD 04 B0  205         CALL MAGIC
AC49 E1        206         POP  HL
AC4A DD E1     207         POP  IX
AC4C D1        208         POP  DE
AC4D C1        209         POP  BC
AC4E E5        210         PUSH HL
AC4F C9        211         RET
AC50           212 ;
AC50           213 SETWIN                       ; (PC) = DATA
AC50 E1        214         POP  HL
AC51 C5        215         PUSH BC
AC52 D5        216         PUSH DE
AC53 DD E5     217         PUSH IX
AC55 11 9D AE  218         LD   DE,MGDATA+1
AC58 06 08     219         LD   B,8
AC5A           220 STWIN2
AC5A 7E        221         LD   A,(HL)
AC5B 23        222         INC  HL
AC5C 12        223         LD   (DE),A
AC5D 13        224         INC  DE
AC5E 10 FA     225         DJNZ STWIN2
AC60 E5        226         PUSH HL
AC61 DD 21 9C  227         LD   IX,MGDATA
AC64 AE
AC65 CD 04 B0  228         CALL MAGIC
AC68 E1        229         POP  HL
AC69 DD E1     230         POP  IX
AC6B D1        231         POP  DE
AC6C C1        232         POP  BC
AC6D E5        233         PUSH HL
AC6E C9        234         RET
AC6F           235 ;
AC6F           236 STSIZE                       ; DE = SIZE
AC6F ED 53 DA  237         LD   (KPSIZE),DE
AC72 AE
AC73 C9        238         RET
AC74           239 ;
AC74           240 STPICH                       ; A = PITCH
AC74 32 DC AE  241         LD   (KPITCH),A
AC77 C9        242         RET
AC78           243 ;
AC78           244 SETDIR                       ; A = DIRECTION
AC78 32 DD AE  245         LD   (KPDIR),A
AC7B C9        246         RET
AC7C           247 ;
AC7C           248 SETLOC                       ; (PC) = POS.
AC7C E1        249         POP  HL
AC7D D5        250         PUSH DE
AC7E 5E        251         LD   E,(HL)
AC7F 23        252         INC  HL
AC80 56        253         LD   D,(HL)
AC81 23        254         INC  HL
AC82 ED 53 DE  255         LD   (KPCURX),DE
AC85 AE
AC86 5E        256         LD   E,(HL)
AC87 23        257         INC  HL
AC88 56        258         LD   D,(HL)
AC89 23        259         INC  HL
AC8A ED 53 E0  260         LD   (KPCURY),DE
AC8D AE
AC8E D1        261         POP  DE
AC8F E5        262         PUSH HL
AC90           263 RSVTBL
AC90 C9        264         RET
AC91           265 ;
AC91           266 ; SUBROUTINES ----------------------
AC91           267 ;
AC91           268 KMSSUB
AC91 C5        269         PUSH BC
AC92 D5        270         PUSH DE
AC93 E5        271         PUSH HL
AC94 CD 16 AE  272         CALL KSTCHK          ; INIT ?
AC97 38 21     273         JR   C,KMSERR
AC99 2A E2 AE  274         LD   HL,(MAGADR)
AC9C 22 95 AE  275         LD   (MADRWK),HL
AC9F 32 5D 1F  276         LD   (#DSK),A
```

▶最近のOh！MZは初心者のためにいろいろと世話をしてくれて素晴しいと思います。用語解説などはとてもうれしいものです。これからも「誰もが読めるOh！MZ」を作ってください。

幅　美貴（16）富山県

```
ACA2                277 KMSSB2
ACA2 CD 1D AE       278         CALL CURCHK             ; CUR. Over ?
ACA5 38 13          279         JR   C,KMSERR
ACA7 DD 56 01       280         LD   D,(IX+1)
ACAA DD 5E 00       281         LD   E,(IX+0)           ; DE <== JIS
ACAD DD 23          282         INC  IX
ACAF DD 23          283         INC  IX
ACB1 7A             284         LD   A,D
ACB2 B3             285         OR   E
ACB3 28 16          286         JR   Z,KMSEND           ; 0000H ?
ACB5 CD D3 AC       287         CALL JISMAG
ACB8 18 E8          288         JR   KMSSB2
ACBA                289 KMSERR
ACBA DD 56 01       290         LD   D,(IX+1)
ACBD DD 5E 00       291         LD   E,(IX+0)
ACC0 DD 23          292         INC  IX
ACC2 DD 23          293         INC  IX
ACC4 7A             294         LD   A,D
ACC5 B3             295         OR   E
ACC6 20 F2          296         JR   NZ,KMSERR
ACC8 37             297         SCF                     ; SET   CF
ACC9 18 04          298         JR   KMSED2
ACCB                299 KMSEND
ACCB CD 50 AE       300         CALL GMAGIC
ACCE B7             301         OR   A                  ; RESET CF
ACCF                302 KMSED2
ACCF E1             303         POP  HL
ACD0 D1             304         POP  DE
ACD1 C1             305         POP  BC
ACD2 C9             306         RET
ACD3                307 ;
ACD3                308 JISMAG
ACD3 CD C4 AD       309         CALL JISADR             ; DE <== RECN
ACD6 38 3C          310         JR   C,CURINC           ; BAD CODE
ACD8 2A 64 1F       311         LD   HL,(#DTBUF)
ACDB 3E 01          312         LD   A,1                ; 1 REC.
ACDD CD 00 20       313         CALL #SCTRD
ACE0 09             314         ADD  HL,BC              ; #DTBUF + OFST
ACE1                315 SETMAG
ACE1 DD E5          316         PUSH IX
ACE3 E5             317         PUSH HL
ACE4 DD E1          318         POP  IX                 ; IX <== KSTADR
ACE6 DD 7E 00       319         LD   A,(IX+0)           ; LINE NO.
ACE9 DD 23          320         INC  IX
ACEB B7             321         OR   A
ACEC 28 24          322         JR   Z,STMGED
ACEE 2A 95 AE       323         LD   HL,(MADRWK)
ACF1 47             324         LD   B,A
ACF2 37             325         SCF                     ; 1ST FLAG
ACF3                326 SETMG2
ACF3 DD 4E 01       327         LD   C,(IX+1)           ; 1ST POS.
ACF6 CD FA AD       328         CALL LCNTCK             ; LINE CONT?
ACF9 C4 8A AD       329         CALL NZ,POSSET
ACFC DD 4E 00       330         LD   C,(IX+0)           ; 2ND POS.
ACFF 79             331         LD   A,C
AD00 32 99 AE       332         LD   (POSWRK),A
AD03 CD 8A AD       333         CALL POSSET
AD06 DD 23          334         INC  IX
AD08 DD 23          335         INC  IX
AD0A 22 95 AE       336         LD   (MADRWK),HL
AD0D CD 46 AE       337         CALL MEMCHK
AD10 10 E1          338         DJNZ SETMG2
AD12                339 STMGED
AD12 DD E1          340         POP  IX
AD14                341 CURINC
AD14 3A DD AE       342         LD   A,(KPDIR)          ; DIRECTION FLAG
AD17 E6 01          343         AND  1                  ; 0:HOR.  1:VER.
AD19 20 12          344         JR   NZ,VERINC
AD1B                345 HORINC
AD1B B7             346         OR   A                  ; RESET CF
AD1C CD 3F AD       347         CALL INCURX             ; INC. X POS.
AD1F CD 1D AE       348         CALL CURCHK             ; CURSOR OVER ?
AD22 D0             349         RET  NC
AD23 21 00 00       350         LD   HL,0
AD26 22 DE AE       351         LD   (KPCURX),HL
AD29 CD 67 AD       352         CALL INCURY             ; INC. Y POS.
AD2C C9             353         RET
AD2D                354 VERINC
AD2D CD 67 AD       355         CALL INCURY             ; INC. Y POS.
AD30 CD 1D AE       356         CALL CURCHK             ; CURSOR OVER ?
AD33 D0             357         RET  NC
AD34 21 00 00       358         LD   HL,0
AD37 22 E0 AE       359         LD   (KPCURY),HL
AD3A 37             360         SCF                     ; SEF CF
AD3B CD 3F AD       361         CALL INCURX
AD3E C9             362         RET
AD3F                363 ;
AD3F                364 INCURX
AD3F F5             365         PUSH AF                 ; CF STORE
AD40 2A DE AE       366         LD   HL,(KPCURX)        ; X POS.
AD43 3A DA AE       367         LD   A,(KPSIZE)         ; CHAR. SIZE
AD46 16 00          368         LD   D,0
AD48 5F             369         LD   E,A
AD49 3A DC AE       370         LD   A,(KPITCH)
AD4C E6 0F          371         AND  0FH                ; SPACE PITCH
AD4E CD 7A AE       372         CALL DEMULA             ; SIZE * PITCH
AD51 CD 69 AE       373         CALL DESR4              ; /16
AD54 3A DA AE       374         LD   A,(KPSIZE)
AD57 CD 8F AE       375         CALL DEADDA             ; + SIZE
AD5A F1             376         POP  AF                 ; CF LOAD
AD5B 38 03          377         JR   C,DCCURX
AD5D 19             378         ADD  HL,DE              ; CASE CY
AD5E 18 03          379         JR   INCRX2
AD60                380 DCCURX
AD60 B7             381         OR   A                  ; CASE NCY
AD61 ED 52          382         SBC  HL,DE
AD63                383 INCRX2
AD63 22 DE AE       384         LD   (KPCURX),HL
AD66 C9             385         RET
AD67                386 INCURY
AD67 2A E0 AE       387         LD   HL,(KPCURY)        ; Y POS.
AD6A 3A DB AE       388         LD   A,(KPSIZE+1)
AD6D 16 00          389         LD   D,0
AD6F 5F             390         LD   E,A
AD70 3A DC AE       391         LD   A,(KPITCH)
AD73 E6 F0          392         AND  0F0H               ; Y PITCH
AD75 0F             393         RRCA
AD76 0F             394         RRCA
AD77 0F             395         RRCA
AD78 0F             396         RRCA                    ; / 16
AD79 CD 7A AE       397         CALL DEMULA             ; PITCH * SIZE
AD7C CD 69 AE       398         CALL DESR4              ; / 1
AD7F 3A DB AE       399         LD   A,(KPSIZE+1)
AD82 CD 8F AE       400         CALL DEADDA             ; + SIZE
AD85 19             401         ADD  HL,DE
AD86 22 E0 AE       402         LD   (KPCURY),HL
AD89 C9             403         RET
AD8A                404 ;
AD8A                405 POSSET
AD8A 16 00          406         LD   D,0                ; X POS.
AD8C 59             407         LD   E,C
AD8D CB 3B          408         SRL  E
```

```
AD8F CB 3B          409         SRL  E
AD91 CB 3B          410         SRL  E
AD93 CB 3B          411         SRL  E                  ; DE = 0-15
AD95 3A DA AE       412         LD   A,(KPSIZE)
AD98 CD 7A AE       413         CALL DEMULA             ; * SIZE
AD9B CD 69 AE       414         CALL DESR4              ; / 16
AD9E E5             415         PUSH HL
AD9F 2A DE AE       416         LD   HL,(KPCURX)        ; + POS.
ADA2 19             417         ADD  HL,DE
ADA3 EB             418         EX   DE,HL
ADA4 E1             419         POP  HL
ADA5 73             420         LD   (HL),E             ; DATA STORE
ADA6 23             421         INC  HL
ADA7 72             422         LD   (HL),D
ADA8 23             423         INC  HL
ADA9 79             424         LD   A,C                ; Y POS.
ADAA E6 0F          425         AND  0FH
ADAC 16 00          426         LD   D,0
ADAE 5F             427         LD   E,A                ; DE = 0-15
ADAF 3A DB AE       428         LD   A,(KPSIZE+1)
ADB2 CD 7A AE       429         CALL DEMULA             ; * SIZE
ADB5 CD 69 AE       430         CALL DESR4              ; / 16
ADB8 E5             431         PUSH HL
ADB9 2A E0 AE       432         LD   HL,(KPCURY)
ADBC 19             433         ADD  HL,DE              ; + POS.
ADBD EB             434         EX   DE,HL
ADBE E1             435         POP  HL
ADBF 73             436         LD   (HL),E             ; DATA STORE
ADC0 23             437         INC  HL
ADC1 72             438         LD   (HL),D
ADC2 23             439         INC  HL
ADC3 C9             440         RET
ADC4                441 ;
ADC4                442 JISADR                          ; DE = JIS CODE
ADC4 EB             443         EX   DE,HL              ; DE <==> HL
ADC5 11 21 21       444         LD   DE,2121H
ADC8 B7             445         OR   A
ADC9 ED 52          446         SBC  HL,DE              ; CODE - 2121
ADCB 3E 5D          447         LD   A,93               ; LOW. < 94 ?
ADCD 95             448         SUB  L
ADCE D8             449         RET  C
ADCF 3E 2E          450         LD   A,46               ; COL. < 47 ?
ADD1 94             451         SUB  H
ADD2 D8             452         RET  C
ADD3 16 00          453         LD   D,0
ADD5 5C             454         LD   E,H
ADD6 3E 5E          455         LD   A,94
ADD8 CD 7A AE       456         CALL DEMULA             ; COL. * 94
ADDB 7D             457         LD   A,L
ADDC CD 8F AE       458         CALL DEADDA             ; + LOW
ADDF 7B             459         LD   A,E
ADE0 E6 03          460         AND  3
ADE2 87             461         ADD  A,A
ADE3 87             462         ADD  A,A
ADE4 87             463         ADD  A,A
ADE5 87             464         ADD  A,A
ADE6 87             465         ADD  A,A
ADE7 87             466         ADD  A,A                ; 64 TIMES
ADE8 06 00          467         LD   B,0
ADEA 4F             468         LD   C,A                ; BC = DATA ADR.
ADEB CB 3A          469         SRL  D
ADED CB 1B          470         RR   E
ADEF CB 3A          471         SRL  D
ADF1 CB 1B          472         RR   E
ADF3 3E 20          473         LD   A,32
ADF5 CD 8F AE       474         CALL DEADDA             ; / 4 + 32
ADF8 B7             475         OR   A                  ; RESET CF
ADF9 C9             476         RET
ADFA                477 ;
ADFA                478 LCNTCK
ADFA 38 0E          479         JR   C,STLCMD
ADFC 3A 99 AE       480         LD   A,(POSWRK)
ADFF A9             481         XOR  C
AE00 20 08          482         JR   NZ,STLCMD
AE02 E5             483         PUSH HL
AE03 2A 9A AE       484         LD   HL,(LNMADR)
AE06 34             485         INC  (HL)
AE07 E1             486         POP  HL
AE08 AF             487         XOR  A                  ; SET ZF
AE09 C9             488         RET
AE0A                489 STLCMD
AE0A AF             490         XOR  A                  ; 'LINE'
AE0B 77             491         LD   (HL),A
AE0C 23             492         INC  HL
AE0D 3E 02          493         LD   A,2                ; 1 LINE
AE0F 77             494         LD   (HL),A
AE10 22 9A AE       495         LD   (LNMADR),HL
AE13 23             496         INC  HL
AE14 B7             497         OR   A                  ; RES ZF
AE15 C9             498         RET
AE16                499 KSTCHK
AE16 3A 94 AE       500         LD   A,(KSTDEV)
AE19 B7             501         OR   A
AE1A C0             502         RET  NZ
AE1B 37             503         SCF                     ; SET CF
AE1C C9             504         RET
AE1D                505 CURCHK
AE1D ED 4B DE       506         LD   BC,(KPCURX)
AE20 AE
AE21 2A A1 AE       507         LD   HL,(MGDATA+5)
AE24 B7             508         OR   A                  ; RESET CF
AE25 ED 42          509         SBC  HL,BC              ; X <  MAX ?
AE27 D8             510         RET  C
AE28 2A 9D AE       511         LD   HL,(MGDATA+1)
AE2B B7             512         OR   A
AE2C ED 42          513         SBC  HL,BC              ; X >= MIN ?
AE2E 28 02          514         JR   Z,CURCK2
AE30 3F             515         CCF
AE31 D8             516         RET  C
AE32                517 CURCK2
AE32 ED 4B E0       518         LD   BC,(KPCURY)
AE35 AE
AE36 2A A3 AE       519         LD   HL,(MGDATA+7)
AE39 B7             520         OR   A
AE3A ED 42          521         SBC  HL,BC              ; Y <  MAX ?
AE3C D8             522         RET  C
AE3D 2A 9F AE       523         LD   HL,(MGDATA+3)
AE40 B7             524         OR   A
AE41 ED 42          525         SBC  HL,BC              ; Y >= MIN ?
AE43 C8             526         RET  Z
AE44 3F             527         CCF
AE45 C9             528         RET
AE46                529 MEMCHK
AE46 ED 5B 97       530         LD   DE,(MAEDWK)
AE49 AE
AE4A EB             531         EX   DE,HL
AE4B B7             532         OR   A
AE4C ED 52          533         SBC  HL,DE
AE4E EB             534         EX   DE,HL
AE4F D0             535         RET  NC                 ; MEMORY OVER?
AE50                536 GMAGIC
AE50 2A 95 AE       537         LD   HL,(MADRWK)
```

▶この感動をどうやって伝えたらいいのでしょうか。とにかくほんとうに懐かしくて涙がこみあげてきました。「ヒーローに休日はない」，なかったのですね，休日が……。
松田　賢一（20）徳島県

```
AE53 36 0F          538          LD   (HL),0FH        ; DONE
AE55 2A E2 AE       539          LD   HL,(MAGADR)
AE58 22 95 AE       540          LD   (MADRWK),HL
AE5B E5             541          PUSH HL
AE5C DD E5          542          PUSH IX
AE5E E5             543          PUSH HL
AE5F DD E1          544          POP  IX
AE61 CD 04 B0       545          CALL MAGIC
AE64 DD E1          546          POP  IX
AE66 E1             547          POP  HL
AE67 37             548          SCF
AE68 C9             549          RET
AE69                550 ;
AE69                551 DESR4
AE69 CB 3A          552          SRL  D                  ; / 16
AE6B CB 1B          553          RR   E
AE6D CB 3A          554          SRL  D
AE6F CB 1B          555          RR   E
AE71 CB 3A          556          SRL  D
AE73 CB 1B          557          RR   E
AE75 CB 3A          558          SRL  D
AE77 CB 1B          559          RR   E
AE79 C9             560          RET
AE7A                561 DEMULA
AE7A E5             562          PUSH HL
AE7B 21 00 00       563          LD   HL,0
AE7E                564 DEMLA2
AE7E B7             565          OR   A
AE7F 28 0B          566          JR   Z,DEMLA4
AE81 CB 3F          567          SRL  A
AE83 30 01          568          JR   NC,DEMLA3
AE85 19             569          ADD  HL,DE
AE86                570 DEMLA3
AE86 CB 23          571          SLA  E
AE88 CB 12          572          RL   D
AE8A 18 F2          573          JR   DEMLA2
AE8C                574 DEMLA4
AE8C EB             575          EX   DE,HL
AE8D E1             576          POP  HL
AE8E C9             577          RET
AE8F                578 DEADDA
AE8F 83             579          ADD  A,E
AE90 5F             580          LD   E,A
AE91 D0             581          RET  NC
AE92 14             582          INC  D
AE93 C9             583          RET
AE94                584 ;
AE94                585 ; WORK EREA ----------------------------
AE94                586 ;
AE94 00             587 KSTDEV:DEFB 0
AE95 00 00          588 MADRWK:DEFS 2
AE97 00 00          589 MAEDWK:DEFS 2
AE99 00             590 POSWRK:DEFS 1
AE9A 00 00          591 LNMADR:DEFS 2
AE9C 00 00 00       592 MGDATA:DEFS 22
AE9F 00 00 00
AEA2 00 00 00
AEA5 00 00 00
AEA8 00 00 00
AEAB 00 00 00
AEAE 00 00 00
AEB1 00
AEB2                593 ;
AEB2                594 MGIDAT
AEB2 06             595          DEFB 06H                 ; WINDOW
AEB3 00 00 00       596          DEFW 0:0:639:199
AEB6 00 7F 02
AEB9 C7 00
AEBB 0A             597          DEFB 0AH                 ; PALET
AEBC 00 07 00       598          DEFB 0:7:0:7:0:7:0:7
AEBF 07 00 07
AEC2 00 07
AEC4 07             599          DEFB 07H                 ; MODE
AEC5 02 00          600          DEFB 2:0
AEC7 0F             601          DEFB 0FH                 ; DONE
AEC8                602 KSTNAM
AEC8 4B 53 54       603          DEFM "KST          .DIC"
AECB 20 20 20
AECE 20 20 20
AED1 20 20 20
AED4 20 2E 44
AED7 49 43
AED9 00             604          DEFB 0
AEDA                605 ;
AEDA 20 10          606 KPSIZE:DEFW 1020H
AEDC 00             607 KPITCH:DEFB 0
AEDD 00             608 KPDIR :DEFB 0
AEDE 00 00          609 KPCURX:DEFW 0
AEE0 00 00          610 KPCURY:DEFW 0
AEE2 00 A0          611 MAGADR:DEFW WRKORG
AEE4 FF AA          612 MADRED:DEFW WRKEND
```

リスト7　ディスクバックアップソースリスト

```
0000                1 ; **************************
0000                2 ; * DISK ALL COPY ROUTINE *
0000                3 ; **************************
0000                4 ;
0000                5 #PRINT EQU  1FF4H
0000                6 #MPRNT EQU  1FE2H
0000                7 #LTNL  EQU  1FEEH
0000                8 #FLGET EQU  2021H
0000                9 #DSK   EQU  1F5DH
0000               10 #SCTRD EQU  2000H
0000               11 #SCTWR EQU  2003H
0000               12 #ETRK  EQU  20FFH
0000               13 ;
0000               14        OFFSET 0C000H-3000H
3000               15        ORG    3000H
3000               16 ;
3000               17 DSKCPY
3000 18 1B         18        JR     DEVST2
3002               19 DEVSET
3002 1A            20        LD     A,(DE)
3003 B7            21        OR     A
3004 28 17         22        JR     Z,DEVST2
3006 CD E3 30      23        CALL   GETDEV
3009 32 1E 31      24        LD     (SOURDV),A
300C CD 07 31      25        CALL   CMPCLN
300F C2 18 31      26        JP     NZ,SYNERR
3012 CD E3 30      27        CALL   GETDEV
3015 32 1F 31      28        LD     (DESTDV),A
3018 CD 07 31      29        CALL   CMPCLN
301B 18 54         30        JR     DSKCP2
301D               31 DEVST2
301D CD E2 1F      32        CALL   #MPRNT
3020 53 4F 55      33        DEFM   'SOURCE      DEVICE ==> '
3023 52 43 45
3026 20 20 20
3029 20 20 20
302C 44 45 56
302F 49 43 45
3032 20 3D 3D
3035 3E 20
3037 00            34        DEFB   0
3038 CD 21 20      35        CALL   #FLGET
303B 32 1E 31      36        LD     (SOURDV),A
303E CD F4 1F      37        CALL   #PRINT
3041 CD EE 1F      38        CALL   #LTNL
3044 CD E8 30      39        CALL   CMPDEV
3047 CD E2 1F      40        CALL   #MPRNT
304A 44 45 53      41        DEFM   'DESTINATION DEVICE ==> '
304D 54 49 4E
3050 41 54 49
3053 4F 4E 20
3056 44 45 56
3059 49 43 45
305C 20 3D 3D
305F 3E 20
3061 00            42        DEFB   0
3062 CD 21 20      43        CALL   #FLGET
3065 32 1F 31      44        LD     (DESTDV),A
3068 CD F4 1F      45        CALL   #PRINT
306B CD EE 1F      46        CALL   #LTNL
306E CD E8 30      47        CALL   CMPDEV
3071               48 DSKCP2
3071 3A 1E 31      49        LD     A,(SOURDV)
3074 CD F4 1F      50        CALL   #PRINT
3077 CD E2 1F      51        CALL   #MPRNT
307A 3A 20 3D      52        DEFM   ': ==> '
307D 3D 3E 20
3080 00            53        DEFB   0
3081 3A 1F 31      54        LD     A,(DESTDV)
3084 CD F4 1F      55        CALL   #PRINT
3087 CD E2 1F      56        CALL   #MPRNT
308A 3A 20 20      57        DEFM   ':  Sure(Y/N)? '
308D 53 75 72
3090 65 28 59
3093 2F 4E 29
3096 3F 20
3098 00            58        DEFB   0
3099 CD 21 20      59        CALL   #FLGET
309C CD F4 1F      60        CALL   #PRINT
309F CD EE 1F      61        CALL   #LTNL
30A2 D6 59         62        SUB    'Y'
30A4 B7            63        OR     A
30A5 C0            64        RET    NZ
30A6               65 DSKCP3
30A6 3E 50         66        LD     A,80
30A8 01 10 00      67        LD     BC,16
30AB 11 00 00      68        LD     DE,0
30AE 21 20 31      69        LD     HL,DSKWRK
30B1               70 CPLOOP
30B1 F5            71        PUSH   AF
30B2 3A 1E 31      72        LD     A,(SOURDV)
30B5 32 5D 1F      73        LD     (#DSK),A
30B8 79            74        LD     A,C
30B9 CD 00 20      75        CALL   #SCTRD
30BC 38 5E         76        JR     C,DSKERR
30BE 3A 1F 31      77        LD     A,(DESTDV)
30C1 32 5D 1F      78        LD     (#DSK),A
30C4 79            79        LD     A,C
30C5 CD 03 20      80        CALL   #SCTWR
30C8 38 52         81        JR     C,DSKERR
30CA EB            82        EX     DE,HL
30CB 09            83        ADD    HL,BC
30CC EB            84        EX     DE,HL
30CD F1            85        POP    AF
30CE 3D            86        DEC    A
30CF 20 E0         87        JR     NZ,CPLOOP
30D1               88 CPYEND
30D1 CD E2 1F      89        CALL   #MPRNT
30D4 43 6F 6D      90        DEFM   'Completed !'
30D7 70 6C 65
30DA 74 65 64
30DD 20 21
30DF 0D 00         91        DEFB   0DH:0
30E1 B7            92        OR     A
30E2 C9            93        RET
30E3               94 ;
30E3               95 GETDEV
30E3 CD 10 31      96        CALL   SPSKIP
30E6 1A            97        LD     A,(DE)
30E7 13            98        INC    DE
30E8               99 CMPDEV
30E8 FE 41        100        CP     'A'
30EA C8           101        RET    Z
30EB FE 42        102        CP     'B'
30ED C8           103        RET    Z
30EE FE 43        104        CP     'C'
30F0 C8           105        RET    Z
30F1 FE 44        106        CP     'D'
30F3 C8           107        RET    Z
30F4 FE 45        108        CP     'E'
30F6 20 0B        109        JR     NZ,CMPDV3
30F8 F5           110        PUSH   AF
30F9 3A FF 20     111        LD     A,(#ETRK)
30FC FE 50        112        CP     80
30FE 20 02        113        JR     NZ,CMPDV2
3100 F1           114        POP    AF
3101 C9           115        RET
3102              116 CMPDV2
3102 F1           117        POP    AF
3103              118 CMPDV3
3103 D1           119        POP    DE
3104 C3 18 31     120        JP     SYNERR
3107              121 CMPCLN
3107 1A           122        LD     A,(DE)
3108 13           123        INC    DE
3109 FE 3A        124        CP     ':'
310B C8           125        RET    Z
310C D1           126        POP    DE
310D C3 18 31     127        JP     SYNERR
3110              128 SPSKIP
```

▶MZ「書院」のクイズは間抜けである。どうせなら，現在出版されているSF雑誌のタイトルを記せ，とか，最近休刊になったアニメ誌はなにか，なんて問題だったらよかったのに（しまった，全部俺の趣味だ……)。　末次　正浩（17）佐賀県

```
3110 1A          129          LD     A,(DE)
3111 13          130          INC    DE
3112 FE 20       131          CP     ' '
3114 28 FA       132          JR     Z,SPSKIP
3116 1B          133          DEC    DE
3117 C9          134          RET
3118             135 SYNERR
3118 3E 0D       136          LD     A,13
311A 37          137          SCF
311B C9          138          RET
```

```
311C             139 DSKERR
311C C1          140          POP    BC
311D C9          141          RET
311E             142 SOURDV
311E 00          143          DEFS   1
311F             144 DESTDV
311F 00          145          DEFS   1
3120             146
3120             147 DSKWRK
```

リスト8　サンプルプログラムソースリスト

```
0000                1 ; **************************
0000                2 ; *  KANJI PRINT SAMPLE    *
0000                3 ; * 1987/06/07  by K.MORI *
0000                4 ; **************************
0000                5 ;
0000                6 KPINIT EQU 0AB00H
0000                7 KPRINT EQU 0AB03H
0000                8 KPCLS  EQU 0AB0FH
0000                9 STSIZE EQU 0AB1BH
0000               10 STPICH EQU 0AB1EH
0000               11 SETWIN EQU 0AB15H
0000               12 SETDIR EQU 0AB21H
0000               13 ;
8000               14        ORG  8000H
8000               15 ;
8000 CD 00 AB      16        CALL KPINIT
8003 D8            17        RET  C
8004 CD 15 AB      18        CALL SETWIN
8007 00 00 00      19        DEFW 0:0:639:191
800A 00 7F 02
800D BF 00
800F 11 20 10      20        LD   DE,1020H       ; YX
8012 CD 1B AB      21        CALL STSIZE
8015 3E 01         22        LD   A,1
8017 CD 21 AB      23        CALL SETDIR
801A 3E 05         24        LD   A,05H          ; YX
801C CD 1E AB      25        CALL STPICH
801F CD 0F AB      26        CALL KPCLS
8022               27 ;
8022 DD 21 40      28        LD   IX,DATA
8025 80
8026 DD 56 01      29 LOOP1: LD   D,(IX+1)
8029 DD 5E 00      30        LD   E,(IX+0)
802C 7A            31        LD   A,D
802D B3            32        OR   E
802E C8            33        RET  Z
802F DD 46 02      34        LD   B,(IX+2)
8032 CD 03 AB      35 LOOP2: CALL KPRINT
8035 13            36        INC  DE
8036 10 FA         37        DJNZ LOOP2
8038 DD 23         38        INC  IX
803A DD 23         39        INC  IX
803C DD 23         40        INC  IX
803E 18 E6         41        JR   LOOP1
8040               42 ;
8040               43 DATA
8040 21 24         44        DEFW 2421H    ; ﾋﾗｶﾞﾅ
8042 53            45        DEFB 83
8043 21 30         46        DEFW 3021H    ; ｶﾝｼﾞ
8045 5E            47        DEFB 94
8046 21 31         48        DEFW 3121H
8048 03            49        DEFB 03H
8049 00 00         50        DEFW 0000H
```

リスト9　ひらがなデータ

●転送プログラム

```
6000 3E 41 32 5D 1F 3E 16 21 : A2
6008 00 80 11 66 00 CD 03 20 : E7
6010 D0 C3 33 20             : E6
----------------------------------
SUM: 0E 84 76 E3 1F 0B 19 41 7682
```

●ひらがなデータ

```
8000 00 00 00 00 00 00 00 00 : 00
8008 00 00 00 00 00 00 00 00 : 00
8010 00 00 00 00 00 00 00 00 : 00
8018 00 00 00 00 00 00 00 00 : 00
8020 00 00 00 00 00 00 00 00 : 00
8028 00 00 00 00 00 00 00 00 : 00
8030 00 00 00 00 00 00 00 00 : 00
8038 00 00 00 00 00 00 00 00 : 00
8040 00 00 00 00 00 00 00 00 : 00
8048 00 00 00 00 00 00 00 00 : 00
8050 00 00 00 00 00 00 00 00 : 00
8058 00 00 00 00 00 00 00 00 : 00
8060 00 00 00 00 00 00 00 00 : 00
8068 00 00 00 00 00 00 00 00 : 00
8070 00 00 00 00 00 00 00 00 : 00
8078 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 00 00 00 00 00 00 00 00 0000

8080 0E 97 47 6B 65 8D 6B 8B : 3F
8088 89 5E 8B 4E 5E 3D 4E 3B : E4
8090 3D 69 3B 99 69 BB 99 BD : F4
8098 BB 9F BD 8F 9F 00 00 00 : 45
80A0 00 00 00 00 00 00 00 00 : 00
80A8 00 00 00 00 00 00 00 00 : 00
80B0 00 00 00 00 00 00 00 00 : 00
80B8 00 00 00 00 00 00 00 00 : 00
80C0 12 34 23 64 34 B2 64 61 : 78
80C8 50 55 61 58 55 6C 58 88 : FF
80D0 86 4D 88 2D 4D 1C 2D 19 : 37
80D8 1C 56 19 86 56 C8 86 CC : 81
80E0 C8 AE CC 7E AE 00 00 00 : 6E
80E8 00 00 00 00 00 00 00 00 : 00
80F0 00 00 00 00 00 00 00 00 : 00
80F8 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 5B D7 BB CE A5 87 C1 51 FF03

8100 05 3B 37 5E 3B 6D 5E BA : 95
8108 A8 BC BA 00 00 00 00 00 : 1E
8110 00 00 00 00 00 00 00 00 : 00
8118 00 00 00 00 00 00 00 00 : 00
8120 00 00 00 00 00 00 00 00 : 00
8128 00 00 00 00 00 00 00 00 : 00
8130 00 00 00 00 00 00 00 00 : 00
8138 00 00 00 00 00 00 00 00 : 00
8140 08 23 12 17 23 19 17 3D : E4
8148 19 59 3D C5 A4 D7 C5 DA : 8E
8150 D7 00 00 00 00 00 00 00 : D7
8158 00 00 00 00 00 00 00 00 : 00
8160 00 00 00 00 00 00 00 00 : 00
8168 00 00 00 00 00 00 00 00 : 00
8170 00 00 00 00 00 00 00 00 : 00
8178 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: A5 73 40 3A 02 5D 3A D1 A2EB

8180 08 86 65 68 49 98 68 A9 : 4D
8188 98 AB A9 9D AB 7F 9D 6F : BF
8190 7F 00 00 00 00 00 00 00 : 7F
8198 00 00 00 00 00 00 00 00 : 00
81A0 00 00 00 00 00 00 00 00 : 00
81A8 00 00 00 00 00 00 00 00 : 00
81B0 00 00 00 00 00 00 00 00 : 00
81B8 00 00 00 00 00 00 00 00 : 00
81C0 0B 71 50 91 71 73 91 47 : 19
81C8 35 75 47 95 75 B7 95 BA : 01
81D0 B7 AC BA 7F AC 6F 7F 00 : 36
81D8 00 00 00 00 00 00 00 00 : 00
81E0 00 00 00 00 00 00 00 00 : 00
81E8 00 00 00 00 00 00 00 00 : 00
81F0 00 00 00 00 00 00 00 00 : 00
81F8 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 16 C3 5F AA 86 B0 AA 19 EDAE

8200 07 86 66 68 49 98 68 3E : E2
8208 98 8C 6B 8E 8C BE 8E 00 : F5
8210 00 00 00 00 00 00 00 00 : 00
8218 00 00 00 00 00 00 00 00 : 00
8220 00 00 00 00 00 00 00 00 : 00
8228 00 00 00 00 00 00 00 00 : 00
8230 00 00 00 00 00 00 00 00 : 00
8238 00 00 00 00 00 00 00 00 : 00
8240 0D 71 50 91 71 82 91 26 : 09
8248 25 65 26 95 65 1D 95 1C : 78
8250 1D 7A 4A 8B 7A 8D 8B 9E : 9C
8258 8D DE 9E 00 00 00 00 00 : 09
8260 00 00 00 00 00 00 00 00 : 00
8268 00 00 00 00 00 00 00 00 : 00
8270 00 00 00 00 00 00 00 00 : 00
8278 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 7B 40 2F A7 25 82 A7 1E 3C94

8280 0C 88 38 6F 66 3D 6F 3C : 89
8288 3D 5A 3C 9A 5A BC 9A BD : DA
8290 BC AE BD 9E AE 8D 9E B8 : 56
8298 A7 00 00 00 00 00 00 00 : A7
82A0 00 00 00 00 00 00 00 00 : 00
82A8 00 00 00 00 00 00 00 00 : 00
82B0 00 00 00 00 00 00 00 00 : 00
82B8 00 00 00 00 00 00 00 00 : 00
82C0 10 25 14 35 25 74 35 50 : 9C
82C8 40 5F 50 3D 5F 1D 3D 97 : 7C
82D0 1D A7 97 C9 A7 CB C9 AD : 0C
82D8 CB 8D AD 7C 8D B3 A3 D5 : 39
82E0 B3 00 00 00 00 00 00 00 : B3
82E8 00 00 00 00 00 00 00 00 : 00
82F0 00 00 00 00 00 00 00 00 : 00
82F8 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 97 48 D9 5E 26 95 85 1A C09F

8300 0C 61 50 48 61 1E 48 16 : E2
8308 15 54 16 74 54 85 74 8A : CA
8310 85 6E 8A 5C 6E C6 A5 EA : 9C
8318 C6 00 00 00 00 00 00 00 : C6
8320 00 00 00 00 00 00 00 00 : 00
8328 00 00 00 00 00 00 00 00 : 00
8330 00 00 00 00 00 00 00 00 : 00
8338 00 00 00 00 00 00 00 00 : 00
8340 0E 61 50 48 61 1E 48 16 : E4
8348 15 54 16 74 54 85 74 8A : CA
8350 85 6E 8A 5C 6E C6 A5 EA : 9C
8358 C6 D4 C3 E2 D1 00 00 00 : 10
8360 00 00 00 00 00 00 00 00 : 00
8368 00 00 00 00 00 00 00 00 : 00
8370 00 00 00 00 00 00 00 00 : 00
8378 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: DA 1A A3 12 17 D2 C2 14 201D

8380 0E 44 34 A3 44 38 27 58 : 24
8388 38 B6 58 60 50 75 60 BB : 86
8390 75 9A BB 4A 9A 2B 4A 2D : 50
8398 2B 3E 2D AE 3E 00 00 00 : 82
83A0 00 00 00 00 00 00 00 00 : 00
83A8 00 00 00 00 00 00 00 00 : 00
83B0 00 00 00 00 00 00 00 00 : 00
83B8 00 00 00 00 00 00 00 00 : 00
83C0 10 44 34 A3 44 38 27 58 : 26
83C8 38 B6 58 60 50 75 60 BB : 86
83D0 75 9A BB 4A 9A 2B 4A 2D : 50
83D8 2B 3E 2D AE 3E C3 B2 D1 : C8
83E0 C0 00 00 00 00 00 00 00 : C0
83E8 00 00 00 00 00 00 00 00 : 00
83F0 00 00 00 00 00 00 00 00 : 00
83F8 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 8E A4 E8 F6 D8 73 54 51 64B6

8400 06 91 80 46 91 47 46 8C : 07
8408 47 9E 8C 9F 9E 00 00 00 : AE
8410 00 00 00 00 00 00 00 00 : 00
8418 00 00 00 00 00 00 00 00 : 00
8420 00 00 00 00 00 00 00 00 : 00
8428 00 00 00 00 00 00 00 00 : 00
8430 00 00 00 00 00 00 00 00 : 00
8438 00 00 00 00 00 00 00 00 : 00
8440 0A 91 80 46 91 47 46 8C : 0B
8448 47 9E 8C 9F 9E A5 94 A6 : 8D
8450 A5 C3 B2 C4 C3 00 00 00 : AI
8458 00 00 00 00 00 00 00 00 : 00
8460 00 00 00 00 00 00 00 00 : 00
8468 00 00 00 00 00 00 00 00 : 00
8470 00 00 00 00 00 00 00 00 : 00
8478 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 43 21 CA 8E 21 33 20 BE 6EBF

8480 0C 23 12 18 23 1A 18 2D : DB
8488 1A 39 2D 86 66 D4 86 A2 : 68
8490 90 AB A2 9D AB 7F 9D 6F : B0
8498 7F 00 00 00 00 00 00 00 : 7F
84A0 00 00 00 00 00 00 00 00 : 00
84A8 00 00 00 00 00 00 00 00 : 00
84B0 00 00 00 00 00 00 00 00 : 00
84B8 00 00 00 00 00 00 00 00 : 00
```

▶誰か，僕の『Oh！MZ 1985年6月号』を知りませんか？　谷川　貴（15）広島県

```
84C0 0E 23 12 18 23 1A 18 2D : DD
84C8 1A 39 2D 86 66 D4 86 A2 : 68
84D0 90 AB A2 9D AB 7F 9D 6F : B0
84D8 7F C2 B1 E1 D0 00 00 00 : A3
84E0 00 00 00 00 00 00 00 00 : 00
84E8 00 00 00 00 00 00 00 00 : 00
84F0 00 00 00 00 00 00 00 00 : 00
84F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 6C D0 73 57 38 DA 76 7C E571

8500 08 53 42 A3 53 75 A3 2A : D5
8508 29 3C 2A 5D 3C BD 5D AC : EE
8510 BD 00 00 00 00 00 00 00 : BD
8518 00 00 00 00 00 00 00 00 : 00
8520 00 00 00 00 00 00 00 00 : 00
8528 00 00 00 00 00 00 00 00 : 00
8530 00 00 00 00 00 00 00 00 : 00
8538 00 00 00 00 00 00 00 00 : 00
8540 0A 53 42 A3 53 75 A3 2A : D7
8548 29 3C 2A 5D 3C BD 5D AC : EE
8550 BD C2 B1 E1 D0 00 00 00 : E1
8558 00 00 00 00 00 00 00 00 : 00
8560 00 00 00 00 00 00 00 00 : 00
8568 00 00 00 00 00 00 00 00 : 00
8570 00 00 00 00 00 00 00 00 : 00
8578 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: DE E0 89 E1 EE 64 00 AC D886

8580 0C 35 24 75 35 C3 75 71 : B8
8588 61 74 71 BA 74 A9 BA 59 : 30
8590 A9 3A 59 3C 3A 5E 3C AE : FA
8598 5E 00 00 00 00 00 00 00 : 5E
85A0 00 00 00 00 00 00 00 00 : 00
85A8 00 00 00 00 00 00 00 00 : 00
85B0 00 00 00 00 00 00 00 00 : 00
85B8 00 00 00 00 00 00 00 00 : 00
85C0 0E 35 24 75 35 A3 75 61 : 8A
85C8 51 64 61 BA 64 99 BA 49 : D0
85D0 99 2A 49 2C 2A 4E 2C AE : 8A
85D8 4E C2 B1 E1 D0 00 00 00 : 72
85E0 00 00 00 00 00 00 00 00 : 00
85E8 00 00 00 00 00 00 00 00 : 00
85F0 00 00 00 00 00 00 00 00 : 00
85F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: BA 68 6D A7 76 54 C6 D0 F495

8600 08 51 30 41 51 4B 41 5D : 04
8608 4B 7E 5D 9E 7E DA 9E D9 : 93
8610 DA 00 00 00 00 00 00 00 : DA
8618 00 00 00 00 00 00 00 00 : 00
8620 00 00 00 00 00 00 00 00 : 00
8628 00 00 00 00 00 00 00 00 : 00
8630 00 00 00 00 00 00 00 00 : 00
8638 00 00 00 00 00 00 00 00 : 00
8640 0C 51 30 41 51 4B 41 5D : 08
8648 4B 7E 5D 9E 7E DA 9E D9 : 93
8650 DA A4 93 A5 A4 C3 B2 C4 : 93
8658 C3 00 00 00 00 00 00 00 : C3
8660 00 00 00 00 00 00 00 00 : 00
8668 00 00 00 00 00 00 00 00 : 00
8670 00 00 00 00 00 00 00 00 : 00
8678 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 21 42 AD 63 42 0D 70 30 7756

8680 10 14 03 E3 14 D2 E3 80 : 53
8688 70 89 80 7A 89 5A 7A 49 : 99
8690 5A 47 49 56 47 76 56 87 : DA
8698 76 8B 89 7D 8B 5F 7D 4F : BD
86A0 5F 00 00 00 00 00 00 00 : 5F
86A8 00 00 00 00 00 00 00 00 : 00
86B0 00 00 00 00 00 00 00 00 : 00
86B8 00 00 00 00 00 00 00 00 : 00
86C0 12 14 03 E3 14 D2 E3 80 : 55
86C8 70 89 80 7A 89 5A 7A 49 : 99
86D0 5A 47 49 56 47 76 56 87 : DA
86D8 76 8B 89 7D 8B 5F 7D 4F : BD
86E0 5F B1 A0 D0 E1 00 00 00 : 61
86E8 00 00 00 00 00 00 00 00 : 00
86F0 00 00 00 00 00 00 00 00 : 00
86F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 60 8F 4A 30 BF 02 60 3E 6EBC

8700 09 17 06 E5 17 43 32 4D : E4
8708 43 5E 4D BE 5E 9B 8A A2 : D1
8710 9B 91 A2 00 00 00 00 00 : CE
8718 00 00 00 00 00 00 00 00 : 00
8720 00 00 00 00 00 00 00 00 : 00
8728 00 00 00 00 00 00 00 00 : 00
8730 00 00 00 00 00 00 00 00 : 00
8738 00 00 00 00 00 00 00 00 : 00
8740 0B 17 06 E5 17 43 32 4D : E6
8748 43 5E 4D BE 5E 9B 8A A2 : D1
8750 9B 91 A2 D3 C2 D0 E1 00 : 14
8758 00 00 00 00 00 00 00 00 : 00
8760 00 00 00 00 00 00 00 00 : 00
8768 00 00 00 00 00 00 00 00 : 00
8770 00 00 00 00 00 00 00 00 : 00
8778 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: D0 0C EA 19 AC 8C 59 DE 528A

8780 0A 52 41 91 52 A2 91 36 : E9
8788 A2 19 36 D6 19 79 97 6B : 5B
8790 79 7D 6B BE 7D 00 00 00 : 9C
8798 00 00 00 00 00 00 00 00 : 00
87A0 00 00 00 00 00 00 00 00 : 00
87A8 00 00 00 00 00 00 00 00 : 00
87B0 00 00 00 00 00 00 00 00 : 00
87B8 00 00 00 00 00 00 00 00 : 00
87C0 0C 52 41 91 52 A2 91 36 : EB
87C8 A2 19 36 D6 19 79 97 6B : 5B
87D0 79 7D 6B BE 7D D3 C2 E1 : 12
87D8 D0 00 00 00 00 00 00 00 : D0
87E0 00 00 00 00 00 00 00 00 : 00
87E8 00 00 00 00 00 00 00 00 : 00
87F0 00 00 00 00 00 00 00 00 : 00
87F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 1C D0 C4 4A D0 09 12 23 1C41

8800 09 74 15 51 41 1E 51 A5 : 38
8808 86 C5 A5 A7 C5 7D 7A 8E : E1
8810 7D DE 8E 00 00 00 00 00 : E9
8818 00 00 00 00 00 00 00 00 : 00
8820 00 00 00 00 00 00 00 00 : 00
8828 00 00 00 00 00 00 00 00 : 00
8830 00 00 00 00 00 00 00 00 : 00
8838 00 00 00 00 00 00 00 00 : 00
8840 0C 74 15 51 41 1E 51 A5 : 3B
8848 86 C5 A5 A7 C5 7D 7A 8E : E1
8850 7D DE 8E C2 B1 C3 C2 E1 : C2
8858 D0 00 00 00 00 00 00 00 : D0
8860 00 00 00 00 00 00 00 00 : 00
8868 00 00 00 00 00 00 00 00 : 00
8870 00 00 00 00 00 00 00 00 : 00
8878 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: EB 2E 90 B2 BD F9 58 47 73C6

8880 0B 34 23 A2 34 60 50 4A : 32
8888 60 59 4A 97 59 B7 97 D9 : 1A
8890 B7 DB D9 AE DB 6E AE 00 : 10
8898 00 00 00 00 00 00 00 00 : 00
88A0 00 00 00 00 00 00 00 00 : 00
88A8 00 00 00 00 00 00 00 00 : 00
88B0 00 00 00 00 00 00 00 00 : 00
88B8 00 00 00 00 00 00 00 00 : 00
88C0 0D 34 23 A2 34 60 50 4A : 34
88C8 60 59 4A 97 59 B7 97 D9 : 1A
88D0 B7 DB D9 AE DB 6E AE D4 : E4
88D8 C3 E2 D1 00 00 00 00 00 : 76
88E0 00 00 00 00 00 00 00 00 : 00
88E8 00 00 00 00 00 00 00 00 : 00
88F0 00 00 00 00 00 00 00 00 : 00
88F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 09 B2 5D CE D0 0A 2A 1A B6D5

8900 05 A8 29 B9 A8 BB B9 8E : 39
8908 BB 5F 8E 00 00 00 00 00 : A8
8910 00 00 00 00 00 00 00 00 : 00
8918 00 00 00 00 00 00 00 00 : 00
8920 00 00 00 00 00 00 00 00 : 00
8928 00 00 00 00 00 00 00 00 : 00
8930 00 00 00 00 00 00 00 00 : 00
8938 00 00 00 00 00 00 00 00 : 00
8940 07 15 04 93 15 B3 93 D5 : E3
8948 B3 D9 D5 8C D9 6C 8C 00 : BE
8950 00 00 00 00 00 00 00 00 : 00
8958 00 00 00 00 00 00 00 00 : 00
8960 00 00 00 00 00 00 00 00 : 00
8968 00 00 00 00 00 00 00 00 : 00
8970 00 00 00 00 00 00 00 00 : 00
8978 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 7A F5 90 D8 96 DA D8 63 3579

8980 09 16 05 94 16 B4 94 D6 : EC
8988 B4 DA D6 8D DA 6D 8D C3 : 88
8990 B2 E2 D1 00 00 00 00 00 : 65
8998 00 00 00 00 00 00 00 00 : 00
89A0 00 00 00 00 00 00 00 00 : 00
89A8 00 00 00 00 00 00 00 00 : 00
89B0 00 00 00 00 00 00 00 00 : 00
89B8 00 00 00 00 00 00 00 00 : 00
89C0 09 14 03 92 14 D2 92 83 : AD
89C8 C2 56 83 48 56 4B 48 6D : 39
89D0 4B BD 6D 00 00 00 00 00 : 75
89D8 00 00 00 00 00 00 00 00 : 00
89E0 00 00 00 00 00 00 00 00 : 00
89E8 00 00 00 00 00 00 00 00 : 00
89F0 00 00 00 00 00 00 00 00 : 00
89F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 85 F9 9F FB 5A 3E FB 89 14B1

8A00 0B 14 03 92 14 D2 92 83 : AF
8A08 C2 56 83 48 56 4B 48 6D : 39
8A10 4B BD 6D B7 A6 D5 C4 00 : 6B
8A18 00 00 00 00 00 00 00 00 : 00
8A20 00 00 00 00 00 00 00 00 : 00
8A28 00 00 00 00 00 00 00 00 : 00
8A30 00 00 00 00 00 00 00 00 : 00
8A38 00 00 00 00 00 00 00 00 : 00
8A40 09 60 50 65 60 77 65 57 : B1
8A48 B6 2A 57 2C 2A 4E 2C 9E : A5
8A50 4E CD 9E 00 00 00 00 00 : B9
8A58 00 00 00 00 00 00 00 00 : 00
8A60 00 00 00 00 00 00 00 00 : 00
8A68 00 00 00 00 00 00 00 00 : 00
8A70 00 00 00 00 00 00 00 00 : 00
8A78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 25 7E 38 22 9A B7 2F E5 683D

8A80 0B 60 50 65 60 77 65 57 : B3
8A88 B6 2A 57 2C 2A 4E 2C 9E : A5
8A90 4E CD 9E B3 A2 D2 C1 00 : A1
8A98 00 00 00 00 00 00 00 00 : 00
8AA0 00 00 00 00 00 00 00 00 : 00
8AA8 00 00 00 00 00 00 00 00 : 00
8AB0 00 00 00 00 00 00 00 00 : 00
8AB8 00 00 00 00 00 00 00 00 : 00
8AC0 0E 63 14 2B 50 B2 92 D4 : 18
8AC8 B2 B4 C3 AB 95 AD AB 9E : 5F
8AD0 AD 7E 9E 6D 7E 6C 6D 7B : 08
8AD8 6C BB 7B DD BB 00 00 00 : 3A
8AE0 00 00 00 00 00 00 00 00 : 00
8AE8 00 00 00 00 00 00 00 00 : 00
8AF0 00 00 00 00 00 00 00 00 : 00
8AF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: E8 A7 35 64 4A 62 FC E2 8B6C

8B00 0B 23 12 19 23 2E 19 3A : FD
8B08 2E 74 63 93 74 C3 93 A4 : 06
8B10 C3 6B 69 8D 6B DD 8D 00 : F9
8B18 00 00 00 00 00 00 00 00 : 00
8B20 00 00 00 00 00 00 00 00 : 00
8B28 00 00 00 00 00 00 00 00 : 00
8B30 00 00 00 00 00 00 00 00 : 00
8B38 00 00 00 00 00 00 00 00 : 00
8B40 18 36 22 48 36 6B 48 81 : 22
8B48 71 75 81 4C 75 3D 4C 2D : DE
8B50 3D 1B 2D 19 1B 27 19 55 : 4E
8B58 27 74 55 B4 74 D6 B4 D9 : 7B
8B60 D6 CC D9 BD CC 9D BD 8C : EA
8B68 9D 8B 8C 9A 8B AA 9A ED : 0A
8B70 AA 00 00 00 00 00 00 00 : AA
8B78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 06 93 68 F1 93 BA F1 33 5787

8B80 11 41 31 4E 41 3D 4E 35 : D2
8B88 15 54 35 1D 54 93 38 B3 : 8D
8B90 93 C4 B3 CC C4 BD CC 8D : B0
8B98 BD 7C 8D 7B 7C 8A 7B AA : 6C
8BA0 8A DD AA 00 00 00 00 00 : 11
8BA8 00 00 00 00 00 00 00 00 : 00
8BB0 00 00 00 00 00 00 00 00 : 00
8BB8 00 00 00 00 00 00 00 00 : 00
8BC0 10 67 74 3C 67 2C 3C 1B : 11
8BC8 2C 18 1B 25 18 52 25 92 : A5
8BD0 52 B3 92 C4 B3 D6 C4 D9 : 81
8BD8 D6 CB D9 BC CB 9D BC 7D : D7
8BE0 9D 00 00 00 00 00 00 00 : 9D
8BE8 00 00 00 00 00 00 00 00 : 00
8BF0 00 00 00 00 00 00 00 00 : 00
8BF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 01 AF 4A 93 D2 08 AE 22 B81A

8C00 10 23 12 19 23 2E 19 3A : 02
8C08 2E 65 54 A5 65 D4 A5 A1 : 0B
8C10 91 AD A1 9E AD 7E 9E 6D : B3
8C18 7E 6C 6D 7B 6C 9B 7B DD : 31
8C20 9B 00 00 00 00 00 00 00 : 9B
8C28 00 00 00 00 00 00 00 00 : 00
8C30 00 00 00 00 00 00 00 00 : 00
8C38 00 00 00 00 00 00 00 00 : 00
8C40 12 23 12 19 23 2E 19 3A : 04
8C48 2E 65 54 A5 65 D4 A5 A1 : 0B
8C50 91 AD A1 9E AD 7E 9E 6D : B3
8C58 7E 6C 6D 7B 6C 9B 7B DD : 31
8C60 9B C2 C1 E2 E1 00 00 00 : E1
8C68 00 00 00 00 00 00 00 00 : 00
8C70 00 00 00 00 00 00 00 00 : 00
8C78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: D2 04 A9 90 23 36 AE 4A B57D

8C80 14 23 12 19 23 2E 19 3A : 06
8C88 2E 65 54 A5 65 D4 A5 A1 : 0B
8C90 91 AD A1 9E AD 7E 9E 6D : B3
8C98 7E 6C 6D 7B 6C 9B 7B DD : 31
8CA0 9B E0 E2 C0 E0 C2 C0 E2 : 61
8CA8 C2 00 00 00 00 00 00 00 : C2
8CB0 00 00 00 00 00 00 00 00 : 00
8CB8 00 00 00 00 00 00 00 00 : 00
8CC0 0A 62 13 35 62 27 35 2C : 9E
8CC8 27 4E 2C 7E 4E AB 7E A4 : 3A
8CD0 AB A4 91 D8 A4 00 00 00 : 5C
8CD8 00 00 00 00 00 00 00 00 : 00
8CE0 00 00 00 00 00 00 00 00 : 00
8CE8 00 00 00 00 00 00 00 00 : 00
8CF0 00 00 00 00 00 00 00 00 : 00
8CF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 8A D5 26 22 D5 AF 4A D7 666E

8D00 0D 62 13 35 62 27 35 2C : A1
8D08 27 4E 2C 7E 4E AB 7E A4 : 3A
8D10 AB A4 91 D8 A4 C2 B1 C3 : 92
8D18 C2 E1 D0 00 00 00 00 00 : 73
8D20 00 00 00 00 00 00 00 00 : 00
8D28 00 00 00 00 00 00 00 00 : 00
```

▶僕は先日,「日本電気が松下・富士通連合軍に脅える理由」という本を読んでNECがとても好きになってしまった。どうしよう。
常世田　一郎（17）埼玉県

```
8D30 00 00 00 00 00 00 00 00 : 00
8D38 00 00 00 00 00 00 00 00 : 00
8D40 0E 62 13 35 62 27 35 2C : A2
8D48 27 4E 2C 7E 4E AB 7E A4 : 3A
8D50 AB A4 91 D8 A4 E1 E3 C1 : E1
8D58 E1 C3 C1 E3 C3 00 00 00 : 0B
8D60 00 00 00 00 00 00 00 00 : 00
8D68 00 00 00 00 00 00 00 00 : 00
8D70 00 00 00 00 00 00 00 00 : 00
8D78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 62 4C 31 F9 6B 47 FA 24 4E73

8D80 0C 93 61 64 93 55 64 57 : 07
8D88 55 8A 57 8C 8A 7D 8C 5D : B2
8D90 7D 1D 4A 0C 1D B9 98 DC : 3A
8D98 B9 00 00 00 00 00 00 00 : B9
8DA0 00 00 00 00 00 00 00 00 : 00
8DA8 00 00 00 00 00 00 00 00 : 00
8DB0 00 00 00 00 00 00 00 00 : 00
8DB8 00 00 00 00 00 00 00 00 : 00
8DC0 10 93 61 64 93 55 64 57 : 0B
8DC8 55 8A 57 8C 8A 7D 8C 5D : B2
8DD0 7D 1D 4A 0C 1D B9 98 DC : 3A
8DD8 B9 C5 B5 D6 C5 D3 C3 E4 : 48
8DE0 D3 00 00 00 00 00 00 00 : D3
8DE8 00 00 00 00 00 00 00 00 : 00
8DF0 00 00 00 00 00 00 00 00 : 00
8DF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 05 39 B9 CE 39 E9 D3 04 C4FC

8E00 10 93 61 64 93 55 64 57 : 0B
8E08 55 8A 57 8C 8A 7D 8C 5D : B2
8E10 7D 1D 4A 0C 1D B9 98 DC : 3A
8E18 B9 D3 D5 B3 D3 B5 B3 D5 : 24
8E20 B5 00 00 00 00 00 00 00 : B5
8E28 00 00 00 00 00 00 00 00 : 00
8E30 00 00 00 00 00 00 00 00 : 00
8E38 00 00 00 00 00 00 00 00 : 00
8E40 05 18 07 54 18 64 54 B9 : 01
8E48 64 EA B9 00 00 00 00 00 : 07
8E50 00 00 00 00 00 00 00 00 : 00
8E58 00 00 00 00 00 00 00 00 : 00
8E60 00 00 00 00 00 00 00 00 : 00
8E68 00 00 00 00 00 00 00 00 : 00
8E70 00 00 00 00 00 00 00 00 : 00
8E78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: B9 0F 97 03 25 A4 8F 1E 6749

8E80 08 18 07 54 18 64 54 B9 : 04
8E88 64 EA B9 C6 B5 C3 B3 D4 : CC
8E90 C3 00 00 00 00 00 00 00 : C3
8E98 00 00 00 00 00 00 00 00 : 00
8EA0 00 00 00 00 00 00 00 00 : 00
8EA8 00 00 00 00 00 00 00 00 : 00
8EB0 00 00 00 00 00 00 00 00 : 00
8EB8 00 00 00 00 00 00 00 00 : 00
8EC0 09 18 07 54 18 64 54 B9 : 05
8EC8 64 EA B9 D3 D5 B3 D3 B5 : EA
8ED0 B3 D5 B5 00 00 00 00 00 : 3D
8ED8 00 00 00 00 00 00 00 00 : 00
8EE0 00 00 00 00 00 00 00 00 : 00
8EE8 00 00 00 00 00 00 00 00 : 00
8EF0 00 00 00 00 00 00 00 00 : 00
8EF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 4F D9 35 41 BA 3E 2E FB F03D

8F00 11 23 12 19 23 2E 19 3A : 03
8F08 2E 66 55 86 66 D4 86 A2 : D1
8F10 72 C1 A2 AD A2 9E AD 7E : ED
8F18 9E 6D 7E 6C 6D 7B 6C AB : F4
8F20 7B DE AB 00 00 00 00 00 : 04
8F28 00 00 00 00 00 00 00 00 : 00
8F30 00 00 00 00 00 00 00 00 : 00
8F38 00 00 00 00 00 00 00 00 : 00
8F40 13 23 12 19 23 2E 19 3A : 05
8F48 2E 66 55 96 66 C4 96 A2 : E1
8F50 72 B1 A2 AD A2 9E AD 7E : DD
8F58 9E 6D 7E 6C 6D 7B 6C AB : F4
8F60 7B DE AB E4 D3 E1 D0 00 : 6C
8F68 00 00 00 00 00 00 00 00 : 00
8F70 00 00 00 00 00 00 00 00 : 00
8F78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 96 1A 64 64 03 07 50 0A 4D8C

8F80 15 23 12 19 23 2E 19 3A : 07
8F88 2E 66 55 86 66 D4 86 A2 : D1
8F90 72 C1 A2 AD A2 9E AD 7E : ED
8F98 9E 6D 7E 6C 6D 7B 6C AB : F4
8FA0 7B DE AB E0 E2 C0 E0 C2 : 28
8FA8 C0 E2 C2 00 00 00 00 00 : 64
8FB0 00 00 00 00 00 00 00 00 : 00
8FB8 00 00 00 00 00 00 00 00 : 00
8FC0 12 24 13 84 24 B3 84 37 : 5F
8FC8 46 48 37 88 48 B7 88 80 : 54
8FD0 70 74 80 7D 74 6E 7D 4E : 8E
8FD8 6E 3D 4E 3C 3D 4B 3C 6B : 64
8FE0 4B 9C 6B BE 9C 00 00 00 : AC
8FE8 00 00 00 00 00 00 00 00 : 00
8FF0 00 00 00 00 00 00 00 00 : 00
8FF8 00 00 00 00 00 00 00 00 : 00
---------------------------------

SUM: 0F 30 77 1B 33 FE 5D 37 387B

9000 10 71 32 4A 71 3B 4A 2B : 1E
9008 3B 1A 2B 19 1A 28 19 57 : 4B
9010 28 77 57 A8 77 DA A8 B5 : 4C
9018 A5 B8 B5 AB B8 7E AB 6E : 0C
9020 7E 00 00 00 00 00 00 00 : 7E
9028 00 00 00 00 00 00 00 00 : 00
9030 00 00 00 00 00 00 00 00 : 00
9038 00 00 00 00 00 00 00 00 : 00
9040 11 73 14 50 40 47 50 3B : FA
9048 47 3D 3B 4E 3D 9E 4E BD : F3
9050 9E B9 BD 27 47 18 27 1A : DB
9058 18 2B 1A 3B 2B C3 B3 E5 : 1E
9060 C3 E6 E5 00 00 00 00 00 : 8E
9068 00 00 00 00 00 00 00 00 : 00
9070 00 00 00 00 00 00 00 00 : 00
9078 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 67 34 74 B6 A9 7B 2E 9C 9B76

9080 10 22 12 48 22 6B 48 91 : F2
9088 80 69 91 3C 69 2C 3C 1B : A2
9090 2C 18 1B 36 18 64 36 A4 : EB
9098 64 D7 A4 DA D7 AD DA 8D : A4
90A0 AD 00 00 00 00 00 00 00 : AD
90A8 00 00 00 00 00 00 00 00 : 00
90B0 00 00 00 00 00 00 00 00 : 00
90B8 00 00 00 00 00 00 00 00 : 00
90C0 0C 60 50 5D 60 6E 5D BE : 02
90C8 6E CD BE CA CD 97 CA 74 : 65
90D0 34 46 74 29 28 3A 29 7A : 1C
90D8 3A 00 00 00 00 00 00 00 : 3A
90E0 00 00 00 00 00 00 00 00 : 00
90E8 00 00 00 00 00 00 00 00 : 00
90F0 00 00 00 00 00 00 00 00 : 00
90F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: B5 ED E4 E4 CF E7 E4 89 5E0F

9100 0B 47 29 A6 47 B7 A6 BA : 7F
9108 B7 9C BA 8C 9C 7B 8C 86 : C2
9110 88 64 86 5B 35 5E 5B 00 : BB
9118 00 00 00 00 00 00 00 00 : 00
9120 00 00 00 00 00 00 00 00 : 00
9128 00 00 00 00 00 00 00 00 : 00
9130 00 00 00 00 00 00 00 00 : 00
9138 00 00 00 00 00 00 00 00 : 00
9140 0D 17 06 93 17 B3 93 D5 : EF
9148 B3 D7 D5 B9 D7 89 B9 78 : A9
9150 89 71 93 61 71 33 42 7C : 50
9158 33 7E 7C 00 00 00 00 00 : 2D
9160 00 00 00 00 00 00 00 00 : 00
9168 00 00 00 00 00 00 00 00 : 00
9170 00 00 00 00 00 00 00 00 : 00
9178 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: C6 24 53 3A 77 FF 1B 09 EF92

9180 0A 3C 35 76 3A A6 76 C8 : 0F
9188 A6 CA C8 AC CA 7C AC 6B : 41
9190 7C 8D 84 6F 8D 00 00 00 : 89
9198 00 00 00 00 00 00 00 00 : 00
91A0 00 00 00 00 00 00 00 00 : 00
91A8 00 00 00 00 00 00 00 00 : 00
91B0 00 00 00 00 00 00 00 00 : 00
91B8 00 00 00 00 00 00 00 00 : 00
91C0 13 24 13 17 24 1A 17 2C : E2
91C8 1A 47 2C 74 47 93 74 B3 : 02
91D0 93 D5 B3 D9 D5 BB D9 7B : D8
91D8 BB 6A 7B 69 6A 81 71 92 : F7
91E0 81 9C 92 7E 9C 5E 7E 00 : A5
91E8 00 00 00 00 00 00 00 00 : 00
91F0 00 00 00 00 00 00 00 00 : 00
91F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 28 D9 80 DC D7 69 75 1F 15A2

9200 0A 7D 74 6E 7D 4E 6E 3D : DF
9208 4E 3C 3D 4B 3C 6B 4B 9C : A0
9210 6B AD 9C 77 A7 00 00 00 : D2
9218 00 00 00 00 00 00 00 00 : 00
9220 00 00 00 00 00 00 00 00 : 00
9228 00 00 00 00 00 00 00 00 : 00
9230 00 00 00 00 00 00 00 00 : 00
9238 00 00 00 00 00 00 00 00 : 00
9240 0B 71 61 7D 71 6E 7D 3E : F4
9248 6E 1C 3E 2B 1C 6B 2B 9C : 41
9250 6B CE 9C A5 B4 75 A5 00 : 48
9258 00 00 00 00 00 00 00 00 : 00
9260 00 00 00 00 00 00 00 00 : 00
9268 00 00 00 00 00 00 00 00 : 00
9270 00 00 00 00 00 00 00 00 : 00
9278 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: A7 C1 88 7D A1 07 06 B3 3648

9280 0C 92 50 73 92 37 45 3B : AA
9288 37 68 3B 87 68 A7 87 C9 : C0
9290 A7 CB C9 AD CB 8E AD 6E : 5C
9298 8E 00 00 00 00 00 00 00 : 8E
92A0 00 00 00 00 00 00 00 00 : 00
92A8 00 00 00 00 00 00 00 00 : 00
92B0 00 00 00 00 00 00 00 00 : 00
92B8 00 00 00 00 00 00 00 00 : 00
92C0 0C 42 31 35 42 39 35 4B : AF
92C8 39 64 4B 73 64 93 73 B5 : 7A
92D0 93 BA B5 AC BA 8E AC 5F : 01
92D8 8E 00 00 00 00 00 00 00 : 8E
92E0 00 00 00 00 00 00 00 00 : 00
92E8 00 00 00 00 00 00 00 00 : 00
92F0 00 00 00 00 00 00 00 00 : 00
92F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: DE 25 85 FB 25 C6 CD D1 C321

9300 10 91 42 A2 91 2A A2 2B : 0D
9308 2A 58 2B 87 58 A7 87 C9 : 83
9310 A7 CC C9 AE CC 6E AE 5D : 2F
9318 6E 5C 5D 6B 5C 7B 6B 8C : 60
9320 7B 00 00 00 00 00 00 00 : 7B
9328 00 00 00 00 00 00 00 00 : 00
9330 00 00 00 00 00 00 00 00 : 00
9338 00 00 00 00 00 00 00 00 : 00
9340 0C 41 30 4E 41 3D 4E 53 : EA
9348 15 1C 53 83 29 A3 83 B4 : 0A
9350 A3 AC B4 BD AC CD BD EB : E1
9358 CD 00 00 00 00 00 00 00 : CD
9360 00 00 00 00 00 00 00 00 : 00
9368 00 00 00 00 00 00 00 00 : 00
9370 00 00 00 00 00 00 00 00 : 00
9378 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 5B 1A CA D0 27 67 D0 CF 4806

9380 0A 91 42 92 91 48 92 2B : 05
9388 48 77 48 A7 77 C9 A7 CC : 61
9390 C9 9E CC 7E 9E 00 00 00 : 4F
9398 00 00 00 00 00 00 00 00 : 00
93A0 00 00 00 00 00 00 00 00 : 00
93A8 00 00 00 00 00 00 00 00 : 00
93B0 00 00 00 00 00 00 00 00 : 00
93B8 00 00 00 00 00 00 00 00 : 00
93C0 09 58 38 5F 56 4E 5F 59 : 54
93C8 2C 78 59 98 78 BA 98 BC : 1B
93D0 BA 9E BC 00 00 00 00 00 : 14
93D8 00 00 00 00 00 00 00 00 : 00
93E0 00 00 00 00 00 00 00 00 : 00
93E8 00 00 00 00 00 00 00 00 : 00
93F0 00 00 00 00 00 00 00 00 : 00
93F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 0A 14 A3 AE 74 19 30 0C 6FA3

9400 0C 52 41 45 52 4E 45 3D : 06
9408 4E 64 16 1C 64 85 38 B5 : BA
9410 85 D7 B5 DA D7 AD DA 8D : D6
9418 AD 00 00 00 00 00 00 00 : AD
9420 00 00 00 00 00 00 00 00 : 00
9428 00 00 00 00 00 00 00 00 : 00
9430 00 00 00 00 00 00 00 00 : 00
9438 00 00 00 00 00 00 00 00 : 00
9440 12 81 42 92 81 3D 92 2D : E4
9448 3D 1C 2D 1B 1C 47 1B 76 : 95
9450 47 B6 76 D8 B6 DB D8 AE : 62
9458 DB 7E AE 6D 7E 6C 6D 7B : 46
9460 6C 8B 7B 9C 8B 00 00 00 : 99
9468 00 00 00 00 00 00 00 00 : 00
9470 00 00 00 00 00 00 00 00 : 00
9478 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 69 E9 1A C9 E9 4B 49 4B 811B

9480 16 81 42 46 81 38 46 65 : 83
9488 46 A5 65 B6 A5 B8 B6 A9 : C2
9490 B8 7A A9 6A 7A 59 6A 68 : EA
9498 59 78 68 89 78 2D 89 1D : 0D
94A0 2D 4B 1D 5B 4B 7E 5B AB : BF
94A8 7E BB AB DD BB 00 00 00 : 7C
94B0 00 00 00 00 00 00 00 00 : 00
94B8 00 00 00 00 00 00 00 00 : 00
94C0 0F 83 23 A2 83 71 60 63 : 0E
94C8 71 37 63 19 37 76 37 87 : 8F
94D0 76 8B 87 D7 D6 88 D7 4C : E0
94D8 88 4D 4C 5E 4D BE 5E 00 : E8
94E0 00 00 00 00 00 00 00 00 : 00
94E8 00 00 00 00 00 00 00 00 : 00
94F0 00 00 00 00 00 00 00 00 : 00
94F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 96 B0 D9 17 FB 21 16 74 60D2

9500 09 71 61 1D 71 78 39 89 : A3
9508 78 8C 89 9D 8C BD 9D DB : EB
9510 BD D9 DB 00 00 00 00 00 : 71
9518 00 00 00 00 00 00 00 00 : 00
9520 00 00 00 00 00 00 00 00 : 00
9528 00 00 00 00 00 00 00 00 : 00
9530 00 00 00 00 00 00 00 00 : 00
9538 00 00 00 00 00 00 00 00 : 00
9540 00 00 00 00 00 00 00 00 : 00
9548 00 00 00 00 00 00 00 00 : 00
9550 00 00 00 00 00 00 00 00 : 00
9558 00 00 00 00 00 00 00 00 : 00
9560 00 00 00 00 00 00 00 00 : 00
9568 00 00 00 00 00 00 00 00 : 00
9570 00 00 00 00 00 00 00 00 : 00
9578 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 3E D6 C5 BA FD 35 D6 64 91AB
```

FM-7/77版 S-OS"SWORD"

Horiuchi Yasuhide
堀内 保秀
Kigoshi Sei
木越 聖

ようこそFMユーザーの皆さん。眠れるZ80カードを呼び起こしS-OSの世界に突入しましょう。Z80カードがなくても大丈夫。ソフトウェアが不可能を可能にしました。リストが多いので注意して入力してください。

FM-7をZ80マシンにする方法

PC-8801, SMC-777, PASOPIAと移植が進められてきていよいよFM-7版の登場です。しかしFM-7ユーザーの方々はきっと首をかしげていることでしょう。そう，6809で動いているFM-7にもともとZ80用のシステムであるS-OSが載ることはないはずなのですから。

6809マシンであるFM-7シリーズでZ80のプログラムを実行させるには2種類の方法が考えられます。ここでカンのいい人はZ80カードを思い出したでしょう。そう，FM-7版"SWORD"はZ80カード上で動くのです。でもカードを持っていない人も安心してください。6809をソフトウェアによってZ80化すること，すなわちZ80エミュレータによってZ80カードのないFM-7でもS-OSの企画に参加できるのです。ただしこの場合，クロックは100KHz程度(！)となっているため，S-OS+「端末をつなぎすぎたミニコン」の世界が一度に楽しめることになってしまいましたが……。

入力方法

まずBASICのモニタなど適当な入力ツールを使って表1のとおりに打ち込んでください。システムモジュールではZ80カード版(以下カード版とする)もエミュレータ版も基本的には同じものですが，一部異なる部分がありますのでエミュレータ版を使用する方は表2にしたがって変更を加えてください。

リスト9にBASIC版のチェックサムプログラムを用意しましたのでこれを使って打ち間違いがないか確かめてください。BASICの処理速度などの都合によりチェックサムはCRCではなく単純に128バイトのデータの和をとった下位8ビット，つまり旧形式のものとしました。次にリスト8のプログラムを走らせてFATとディレクトリ部分を作るのですが，安全のため，

SAVEM"SOSbak",&H2100,&H5AFF,0

とセーブしてからリスト8を打ち込んで，間違いがないかよく確かめてから実行してください。打ち間違いがあるとせっかく打ち込んだデータが破壊されてしまうことがあります。その場合は"SOSbak"をロードしてもう一度やり直してください。データの破壊などが起きてないようでしたら，

SAVEM"SOSo",&H2100,&H5AFF,0

とセーブしてファイル"SOSo"の作成は完了します。

なお，"SOSo"やこのあと出てくるプログラムはすべて1枚のディスクに納めるようにしてください。

システムディスクの作成

まず，リスト7のシステムジェネレータを打ち込んでセーブしてください。そしてリスト6のダンプリストも打ち込んでリスト9のチェックサムプログラムで打ち間違いがないか確かめてから，

SAVEM"SYSGENo",&H2000,&H2079,&H2000

でセーブしておきます。

ここまでできたら，以上すべてのリストをセーブしたディスクをドライブ0に入れてリスト7のプログラムを走らせます。そしてメッセージがでたら，ドライブ0にS-OSシステムディスクにする物理フォーマット済みのディスクを入れてドライブを指定し"Y"を入力すれば，システムディスクが作成されます。

このとき，メモリエディタを組み込むかどうか聞いてきますので，エミュレータ版で初めてシステムジェネレートするときは，必ずメモリエディタを組み込むように指定してください。また，何枚かシステムディスクを作りたいときは続けて作成することができます。

物理フォーマットにはFMのシステムディスクについてくる"SYSDSK"ユーティリティなどを使ってください。一度システムディスクを作ったあとはS-OS上からFORMAT&SYSGENを使ってフォーマット(論理フォーマットのみ)を行ってください。

表2　変更点

アドレス	Z80カード版	エミュレータ版
387AH	B8H	BDH
3887H～3888H	0893H	103DH
38B8H	3BH	39H
4033H～4047H	リスト2による	
4080H～4AFDH	リスト3による	
4B20H～4B21H	7516H	
4E50H～4E51H	8010H	0021H
4F8FH～4F90H	8010H	

表1　入力部分

アドレス	内容
2100H～21FAH	IPL
2200H～33FFH	FAT，ディレクトリ(リスト8のプログラムで作成する)
3800H～5B18H	システム (5100H～5AFFHまではDOSモジュール)

S-OSの使い方

S-OSというのは基本ソフトウェアです。S-OS“SWORD”を入力したからといって，それだけではほとんどなにもできません。“SWORD”はあくまで前提条件，S-OSの世界はS-OS用のアプリケーションによって開かれるのです。過去2年間，毎月のように発表されてきたS-OS用のアプリケーションがFM-7/77シリーズでもそのまま使うことができます。

ユーザーは表7に挙げられたモニタコマンドを使って（今回の“SWORD”にはRAMディスクやバッチ処理は組み込まれていない），表9の内部サブルーチンを使って作られたアプリケーションを走らせるのです。

アプリケーションを入力するにはカード版はデバッガを，エミュレータ版はメモリエディタを使用してください。デバッガのコマンドは表3，メモリエディタのコマンドは表8にまとめられています。なお，メモリエディタは3000Hから置かれていますのでアドレスの重なるものはアドレスをずらして入力してください。

メモリエディタでは3099Hを書き換えることにより（メモリエディタ自身で書き換え可）CRCチェックバイトを出力することが可能です。計算にやや時間がかかりますが，入力のチェックに役立てることができるでしょう。

メモリマップ

メモリマップは図1のようになっています。6809とZ80でデータが競合しないように，それぞれ専用のスタックを持たせています。800Hから6809システムモジュールが置かれ大部分の処理はこの部分で行われることになります。1080Hからはカード版ではデバッガが，エミュレータ版ではエミュレータが入っています。

1B00HからがZ80システムモジュールで，ここは6809部分を呼び出すのが主な仕事です。2100Hからは全機種共通のDOSモジュールが置かれ，2B00HからはディスクI/O，2E00HからはFATバッファとセクタバッファになっています。3000HからFBFFHまでがフリーエリアです。

カード版の動作

IPLによりプログラムがロードされると，まず6809システムモジュールが呼び出されます。そこでZ80および6809の割り込みテーブルの初期化やシステムの初期化を行ったあと6809を停止させZ80を起動するのです。最初にZ80が呼び出されたときは0000H番地から実行が開始され（リセット），そこからコールドスタートへジャンプします。ここではZ80スタックポインタの初期化やスクリーンのモード設定をしたあとコマンド待ちの状態に入ります。

画面モードの設定や，ディスクアクセスなどの実行は6809（BIOS）が受け持ちます。Z80からBIOSがコールできるようにシステムコールを用意しました。RST 08Hに続けてリクエスト番号を書いておくと，その番号のワークエリア（FUNCNO）にストアしてから，Z80を停止させ6809を再起動しシステムコール実行ルーチンが走り，ジョブが完了すると再びZ80が起動され6809が停止します。システムコールではパラメータの受け渡しはすべてZ80のレジスタにより行われ，Z80から6809に制御が移る際Z80のレジスタを裏レジスタを含めてすべてスタックに積み，そのスタックポインタを6809に渡すことによりレジスタを参照させ，また結果を返させています。

このシステムコールを利用することによりZ80で書かれている部分（SWORD本体）は非常にコンパクトになっています。

エミュレータ版の動作

エミュレータ版の動作は，基本的にカード版と同じですが，エミュレータの性質上若干の違いがあります。ここではその違いについてだけ説明します。

まずRST 08Hが検出されるとカード版と同様すべてのレジスタをスタックに積むのですが，カード版のようにZ80のPUSH命令を使うわけにはいかないので6809で2バイトずつ上位と下位を逆転させながら積んでいきます（もともとメモリ上にある仮想レジスタをわざわざ上下逆にして転送するのはスピードの低下の一因となっているのですが，なにぶんカード版をそのまま利用するということでこのようにしています）。このあとカード版ではZ80を再起動させていましたが，エミュレータ版ではシステムコール実行ルーチンを直接コールし，帰ってきたところで再びエミュレートを開始するようにして同様の動作ができるようにしてあります。

エミュレータの動作

エミュレータではZ80の裏レジスタを含むすべてのレジスタをメモリ上に持っています。すなわち命令にしたがって仮想レジスタを更新していくことがエミュレータの動作というわけです。実際のエミュレータの動作は次のように進められます。

1)　1バイトフェッチする（読み出す）
2)　命令コードを分類し，各ルーチンを呼び出す
3)　1)へ戻る

では，それぞれについて説明していきましょう。

1)ここは単純で，仮想プログラムカウンタのさすメモリから1バイトフェッチします。また，ここではコードの解析のために，|b_7b_6|$b_5b_4b_3$|$b_2b_1b_0$|のように3つのビットフィールドに分けたデータも用意します。これらのデータを説明の都合上それぞれ上位からDATA1，DATA2，DATA3と呼ぶことにします。

2)ここがエミュレータのメインになる部分でいちばん時間のかかる部分でもあります（ま，実際エミュレートしてるのがこの部分ですから当然のことですけど）。図2を見るとわかるようにZ80のコードは（Z80に限ったことではありませんが）ただでたらめに並んでいるのではなく，ある一定の規則にしたがって並んでいます。

まずDATA1によってコードを4のブロックに分けます。第2ブロックでは，DATA2がデスティネーションとなるレジスタを，DATA3がソースとなるレジスタあるいはアドレッシングモードを示すので，これにしたがって仮想レジスタを書き換えます。第3ブロックではDATA2が演算の種類を，DATA3が演算対象となるレジスタを示します。第1ブロック，第4ブロックでは8バイトおきに類似した命令が並んでいるので，DATA3によって大まかに命令を分けたあと，

DATA2で決定します。Z80ではこのほかにも2バイト系，3バイト系の命令がありますが，以上と同様に解析していきます。

注意点

このエミュレータではスピードの低下を避けるため，コードのチェックをしていません。このため定義されていないコードを実行しようとした場合の動作は保証されません（まあ，本来どんなCPUでも保証されていないところですが)。ただし，有名な未定義命令（IX, IYをHLレジスタのように8ビットごとに分けて扱う）はきっとうまく動作してくれるはずです，が，保証の限りではありません。また，IN/OUTのようなポートに関する命令やDI/EI/IM 0のような割り込み関係の命令，「LD A, R」のようなわけのわからない命令はすべてNOPと同様の動作をします。ただし，RETI などは単にリターンするだけになっています。

6809では「SUB命令ではハーフキャリフラグが定義されていない」，「論理演算のときパリティチェックをしない」，となっているので，「SUBを実行する前と後ろのデータを比較してハーフキャリを立てる」，「1ビットずつシフトして立っているビット数を数える」などかなり無駄な処理（実際これらのフラグが頻繁に使われるプログラムをいまだに見たことがないのだが，しかたがないのでやっている）をしなくてはならなかったことやいきあたりバッタリに作っていったことでコード解析に無駄があり，かなり遅いものに仕上がっています。

しかし，改良しだいでもっと速くすることができるでしょうし，エミュレータだけを差し換えて使うことも簡単ですから（ただしサイズが問題になる）この遅さにめげてしまった人は自分でエミュレータ部分を作り変えてみてはいかがでしょうか。もともとのカード版のほうもI/O関係の処理を6809にまかせているためタスクの切り換えに時間がかかっています。この部分も改善の余地は残されています（これは読者の皆さんへのひそかな挑戦でもあります)。

図1メモリマップ

図2　Z80のコード

00 〜 3F	40〜7F		80〜BF		C0 〜 FF
各種命令	40	LD B,?	80	ADD	各種命令
	48	LD C,?	88	ADC	
	50	LD D,?	90	SUB	
	58	LD E,?	98	SBC	
	60	LD H,?	A0	AND	
	68	LD L,?	A8	XOR	
	70	LD (HL),?	B0	OR	
	78	LD A,?	B8	CP	

他機種版との違い

S-OSのコマンドに#M(各機種のモニタのホットスタートにジャンプする）というのがありますが，Z80上で動くモニタが必要になるためカード版ではデバッガを搭載しました。内容はコマンド一覧表に示すように「各機種用モニタ」に比べればかなり高機能といえるでしょう。

ただしエミュレータ版ではデバッガの入っていたところにエミュレータを組み込んだためモニタがありません。このため#Mコマンドを入力してもS-OSのホットスタートに飛ぶだけです。この関係でカード版ではシステムコールのとき不正なリクエストナンバーを指定するとシステムコールエラーが発生しデバッガへ制御が移りますが，エミュレータ版ではホットスタートへ飛ぶだけになっています。

また，FM-7版ではテープをサポートしていません。まだディスクI/O部分に余裕があるので力のある人は拡張してみてください（ソースを見ればわかると思いますが，HLレジスタにアドレスを入れて6809のプログラムを実行させるシステムコールがあるのでそれを利用することになるでしょう)。

ついでに，サブルーチン#GETKY（1FD0H）が完全なリアルタイムキー入力には

なっていません。これはFM-7のキー入力の性質によるものですがエミュレータの高速化，テープへの対応とあわせて投稿を期待しています。

最後に

いよいよZ80以外のCPUを積んだマシンにもS-OSが移植されました。CPUが違うからといってあきらめることはありません。機種の壁を越えるというのがS-OSの基本思想なのですから，あらゆる機種がこの試みに参加してしかるべきであるといえるでしょう。

今回の移植は6809へのものでしたが，エミュレータを使うことで8086や68000などの16ビット機でもS-OSを走らせることが可能です。つまりメディアの違い（2Dと2HD）さえ乗り越えれば，PC-9801やX68000ともプログラムやデータの共通化が可能になるのです。エミュレータという方法をとるかぎり処理速度は低下してしまいますが，その先にはもっと大事なものが待っているはずです。全国のFMユーザーの皆さん，ともにこの素晴しい試みに参加しようではありませんか。

表3　デバッガのコマンド一覧（カード版のみ）

コマンド	機能
B	現在のブレイクポイントの表示
B*	ブレイクポイントの解除
B adr	adr1にブレイクポイントを設定する
C adr1, adr2	adr1＋adr2, adr1－adr2を計算する
D[adr1, [, adr2]]	adr1から(adr2までを)ダンプする
F adr1, adr2, dt	adr1からadr2までをdtで埋める(dtは1バイトのデータ)
G[adr1]	Rコマンドの設定にしたがってプログラムを実行する
H	ヘルプメニューの表示
I adr1, str	abr1からstrを書き込む(strは文字列)
J adr1	adr1へジャンプする
M[adr1]	adr1からのメモリを変更する
P	プリンタへの出力ON/OFF
Q	S-OSへ戻る
R	レジスタの表示・変更(変更は表示された値を書き換える)
S adr1, adr2, dt	adr1からadr2の範囲でdtをサーチする
T adr1, adr2, adr3	adr1からadr2までをadr3へ転送する

表4　S-OSアスキーコード

下位＼上位	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	nul		SP	0	@	P		p					タ	ミ		
1			!	1	A	Q	a	q			。	ア	チ	ム		
2			"	2	B	R	b	r			「	イ	ツ	メ		
3			#	3	C	S	c	s			」	ウ	テ	モ		
4			$	4	D	T	d	t				エ	ト	ヤ		
5			%	5	E	U	e	u				オ	ナ	ユ		
6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7			'	7	G	W	g	w			ァ	キ	ヌ	ラ		
8			(	8	H	X	h	x			ィ	ク	ネ	リ		
9			)	9	I	Y	i	y			ゥ	ケ	ノ	ル		
A			*	:	J	Z	j	z			ェ	コ	ハ	レ		
B		BRK	+	;	K	[	k	■			ォ	サ	ヒ	ロ		
C	CLS	→	,	<	L	＼	l				ャ	シ	フ	ワ		
D	CR	←	−	=	M	]	m				ュ	ス	ヘ	ン		
E		↑	.	>	N	^	n				ョ	セ	ホ	゛		
F		↓	/	?	O		o	π			ッ	ソ	マ	゜		

表5　デバイス名

A	SWORD形式のディスクドライブ	A
B	〃	B
C	〃	C
D	〃	D
E～L	リザーブ	
T	共通フォーマットテープ(未使用)	
S	システムフォーマットテープ(未使用)	
Q	リザーブ	

表6　エラーメッセージ

No.	メッセージ	内容
1	Device I/O Error	入出力時にエラーが発生した
2	Device Offline	デバイスがつながっていない
3	Bad File Descriptor	ファイルディスクリプタが間違っている
4	Write Protected	ライトプロテクトがかかっている
5	Bad Record	レコードナンバーに間違いがある
6	Bad File Mode	アトリビュートが違う
7	Bad Allocation Table	ファットエラー
8	File not Found	ファイルが見つからない
9	Device Full	ディスクがいっぱい
10	File Already Exists	すでに同名のファイルが登録されている
11	Reserved Feature	現在使用されていない
12	File not Open	ファイルをオープンせずに読み書きしようとした
13	Syntax Error	文法間違い
14	Bad Data	正しい引き数ではない

表7　S-OS"SWORD"モニタコマンド
（[　]は省略可能であることを示す）

#D[<デバイス名>：]
<デバイス名>で指定されたデバイスのディレクトリを表示する。省略時はデフォルトのディレクトリ。

#DV <デバイス名>：
デフォルトデバイスを変更する。

#J <アドレス>
アドレスから始まるプログラムをコールする。サブルーチン中のRETでS-OSのモニタにリターンできる。

#K <ファイル名>
<ファイル名>で与えられたファイルを消去する。

#L <ファイル名>[：<ロードアドレス>]
<ファイル名>で与えられたファイルを<ロードアドレス>へロードする。ロードアドレスが省略されたときには，セーブしたときのアドレスへロードする。

#M
各機種のマシン語モニタのホットスタートへジャンプする。

#N <ファイル名1>：<ファイル名2>
<ファイル名1>を<ファイル名2>に変更する。なお，<ファイル名2>のデバイス指定は不要。

#S<ファイル名>：<開始番地>：<終了番地>[：<実行番地>]
<開始番地>から<終了番地>までを <ファイル名> でセーブする。

#ST <ファイル名>：P　または　：R
<ファイル名>で指定されたファイルにライトプロテクトをかける。その後は同一ファイルをセーブ，消去ができなくなる。プロテクトをはずすにはRを指定。

#W
画面の40字，80字モードを切り換える。

#!
ブートコマンド。

表8　メモリエディタのコマンド一覧

T：128バイト前へ
G：128バイト後ろへ
R：縦サム計算・表示

BREAK：アドレス入力に戻る

注1）アドレス入力時にブレイクするとS-OSに戻る
注2）3099HのC9Hを00Hにすると，縦サムとCRCを表示するようになる

表9 S-OSサブルーチン

ルーチン名(アドレス)	サブルーチンの機能	レジスタ破壊
#COLD (1FFDH)	S-OSのコールドスタート。初期設定後メッセージを出力し，ワークエリアUSRに格納されているアドレスにジャンプする。USRには初期値として#HOTのアドレスが格納されている。	—
#HOT (1FFAH)	S-OSのモニタになっており，プロンプト#が出てコマンド入力待ちになる。	—
#VER (1FF7H)	HLレジスタにS-OSの機種とバージョンを返す。Hレジスタは機種を表しており，上位4ビットで機種の系列を示し，下位4ビットで系列内の機種番号を示す。 0 0 MZ-80K/C/1200 0 1 MZ-700 0 2 MZ-1500 1 0 MZ-80B 1 1 MZ-2000/2200 1 2 MZ-2500 2 0 X1/C/D/F/G/turbo 3 0 PC-8801/mkII/SR/TR/FR/MR/FH/MH 4 0 FM-7/77 5 0 SMC-777 6 0 PASOPIA LレジスタはS-OSのバージョンNo.を示しており，各種パッケージを追加したりした場合のS-OSのバージョンをチェックできるようにする。	HL
#PRNT (1FF4H)	Aレジスタの内容をアスキーコードとみなし表示する（1文字表示)。	F
#PRNTS (1FF1H)	スペースをひとつ表示する。	F
#LETNL (1FEEH)	改行する。	なし
#NL (1FEBH)	カーソルが行の先頭になければ改行する。	なし
#MSG (1FE8H)	DEレジスタの示すアドレスから0DHがあるまでアスキーコードとみなし文字列表示する。	F
#MSX (1FE5H)	DEレジスタの示すアドレスから00Hがあるまでアスキーコードとみなし文字列表示する。	F
#MPRNT (1FE2H)	これをコールした次のアドレスから00Hがあるまでアスキーコードとみなし文字列表示する。 例) CALL #MPRNT DM "MESSAGE" DB 0	AF DE
#TAB (1FDFH)	Bレジスタの値とカーソルX座標との差だけスペースを表示する。	AF
#LPRNT (1FDCH)	Aレジスタの内容をアスキーコードとみなしプリンタのみに出力する。プリンタエラーがあった場合は，キャリフラグをセットしてリターンする。	AF
#LPTON (1FD9H)	上記#PRINT～#TAB,#PRTHX, #PRTHLの出力をディスプレイだけでなくプリンタにも出力するかどうかのフラグ#LPTSWをセットする。これをコールしたあとは，上記サブルーチンでプリンタにも出力される。	なし
#LPTOF (1FD6H)	フラグ#LPTSWをリセットする。これをコールしたあとは，#PRINT～#TAB, #PRTHX, #PRTHLの出力をディスプレイのみにする。	なし
#GETL (1FD3H)	DEレジスタにキー入力バッファの先頭アドレスを入れてコールすると，キーボードから1行入力をして文字列をバッファに格納しリターンする。エンドコードは00H。途中でSHIFT+BREAKが押されたら，バッファ先頭に1BHが格納される。	AF
#GETKY (1FD0H)	キーボードからリアルタイムキー入力をする。入力したデータはAレジスタに格納され，何も押されていないときはAレジスタに0をセットしてリターンする。	AF
#BRKEY (1FCDH)	ブレイクキーが押されているかどうかをチェックする。押されているときはゼロフラグをセットしてリターンする。	AF
#INKEY (1FCAH)	何かキーを押すまでキー入力待ちをし，キー入力があるとリターンする。押されたキーのアスキーコードはAレジスタにセットされる。	AF
#PAUSE (1FC7H)	スペースが押されていれば，再び何かキーを押すまでリターンしない。このときSHIFT+BREAKを押すと，このルーチンをコールした次のアドレスの2バイトの内容を参照し，そこへジャンプする。 例) CALL #PAUSE DW BRKJOB ここでブレイクキーを押すとBRKJOBへジャンプ さもなくばDW BRKJOBはスキップ。	AF
#BELL (1FC4H)	ベル（ビープ音）を鳴らす。	AF
#PRTHX (1FC1H)	Aレジスタの内容を16進数2桁で表示する。	AF
#PRTHL (1FBEH)	HLレジスタの内容を16進数4桁で表示する。	AF
#ASC (1FBBH)	Aレジスタの下位4ビットの値を16進数を表すアスキーコードに変換し，Aレジスタにセットする。	AF
#HEX (1FB8H)	Aレジスタの内容を16進数を表すアスキーコードとしてバイナリに変換し，Aレジスタにセットする。Aレジスタの内容が16進数を表すアスキーコードでない場合は，キャリフラグをセットしてリターンする。	AF
#2HEX (1FB5H)	DEレジスタの示すアドレスから2バイトの内容を，2桁の16進数を表すアスキーコードとしてバイナリに変換し，Aレジスタにセットする。エラーの場合はキャリフラグがセットされる。	AF DE+2
#HLHEX (1FB2H)	DEレジスタの示すアドレスから4バイトの内容を，4桁の16進数を表すアスキーコードとしてバイナリに変換し，HLレジスタにセットする。エラーがあった場合は，キャリフラグがセットされる。	AF HL DE+4
#WOPEN (1FAFH)	#FILEでセットされたファイル名，(#DTADR)，(#SIZE)，(#EXADR)をテープに書き込む。ディスクの場合は，新しいファイルかどうかのチェックを行う。エラー発生時にはキャリフラグが立つ。	AF BC DE HL
#WRD (1FACH)	(#DTADR)，(#SIZE)，(#EXADR)に従って，デバイスにデータをセーブする。ディスクの場合#WOPEN後でないとFile not Openのエラーが出る。	〃
#FCB (1FA9H)	テープの場合従来の#RDIとまったく同じ。ディスクの場合#DIRNOの値に従って(#IBFAD)にディレクトリの内容を転送する。これにより従来のテープロードルーチンにまったく手を加えることなくディスクリードを行うことができる。CALL後，(#DIRNO)はインクリメントされる。ブレイクキーが押されると(#DIRNO)をクリアする。リターンキーが押されるとキャリフラグを立ててリターンする。	〃
#RDD (1FA6H)	(#DTADR)，(#SIZE)，(#EXADR)に従って，デバイス上のファイルを読み込む。#ROPEN後でないとFile not Openのエラーが出る。	〃
#FILE (1FA3H)	Aレジスタにファイルのアトリビュート，DEレジスタにファイル名の入っている先頭アドレスをセットしてコールすると(#IBFAD)にファイル名のセットと(#DSK)にファイルディスクリプタのセットを行う。ファイルを操作する前には，必ずこのサブルーチンにより，ファイル名とアトリビュートをセットしなければならない。コール後DEレジスタは行の終わり(00H)か：(コロン)の位置を示している。	〃
#FSAME (1FA0H)	#FILEでセットされたファイルネームと，読み込んだファイルネームを比較する。一致すればゼロ，不一致ならばノンゼロでリターンする。アトリビュートのチェックも同時に行う。	〃
#FRRNT (1F9DH)	テープから読み込んだファイルネームを表示する。スペースキーを押すと表示後一時停止する。	〃
#POKE (1F9AH)	HLレジスタの内容をオフセットアドレスとして，CIOS用特殊ワークエリアにAレジスタの内容を書き込む。	なし
#POKE@ (1F97H)	メインメモリからS-OS用特殊ワークエリアにデータを転送する。HLレジスタにメモリ先頭アドレス，DEレジスタにワークエリアオフセットアドレス，BCレジスタにバイト数を入れてコールする。	AF BC DE HL
#PEEK (1F94H)	HLレジスタの内容をオフセットアドレスとして，S-OS用特殊ワークエリアからAレジスタにデータを読み出す。#POKEと逆の動作。	AF
#PEEK@ (1F91H)	S-OS用特殊ワークエリアからメインメモリにデータを転送する。HL，DE，BCレジスタにセットするパラメータは#POKE@と同じ。	AF DE BC HL

#MON (1F8EH)	各機種のモニタにジャンプする。	—
[HL] (1F81H)	HLレジスタにコールしたいアドレスを入れ， CALL [HL] と使うことにより，擬似的な間接コールが可能。	なし
#GETPC (1F80H)	現在のプログラムカウンタの値をHLにコピーする。	HL

#DRDSB (2000H)	DEが示すレコードナンバーからAが示すレコード数だけHLが示すアドレスに読み込む。連続セクタリード。(#DSK)にデバイス(A～D)をセットしてコールする。
	LD DE,(#FATPOS)
	LD HL,(#FATBF)
	LD A,1
	CALL #SCTRD
	とすれば，FATバッファにFATを読み出すことができる。
		AF
		AF

#DWTSB (2003H)	HLが示すアドレスからAレコード分(A×256バイト)の内容を，DEを先頭レコードとして記録する。連続セクタライト。(#DSK)にデバイス(A～D)をセットしてコール。	AF AF
#DIR (2006H)	(#DSK)で指定されたデバイス上の全ディレクトリを表示する。	AF BC DE HL
#ROPEN (2009H)	テープの場合は，先に#FILEでセットされたファイル名と，読み込んだIBを比較し，同一ファイルならゼロ，違えばノンゼロでリターンする。ディスクの場合は，#FILEでセットされたファイルがディスク上にあるかどうかのチェックを行う。ゼロフラグは常にリセットとなる。いずれの場合にも，エラーが発生したときにはキャリでリターンする。またファイルの情報は，(#DTADR),(#SIZE),(#EXADR)へ転送される。	〃
#SET (200CH)	#IBFADで示されるIBバッファの内容と一致するディスク上のファイルをライトプロテクトする。	〃
#RESET (200FH)	#IBFADで示されるIBバッファの内容と一致するファイルのライトプロテクトをはずす。	〃
#NAME (2012H)	#FILEで設定されたファイル名を，DEレジスタが示すメモリ上のデータに変える。リネーム。メモリ上のデータ中にデバイスディスクリプタが入っていても無視する。またDE+16以内にエンドコード(00H，':')がないときにはエラーが発生する。	〃
#KILL (2015H)	#IBFADで示されるIBバッファの内容と一致するディスク上のファイルをキルする。	〃
#CSR (2018H)	現在のカーソル位置を，HにY座標，LにX座標の順で読み出す。以後，カーソル位置の読み出しは必ずこの方法によること。(#XYADR)は使わない。	HL
#SCRN (201BH)	HにY座標，LにX座標をセットしコールすると，画面上の同位置にあるキャラクタをAに読み出す。	AF
#LOC (201EH)	HにY座標，LにX座標を入れてコールすると，カーソル位置がそこにセットされる。以後，カーソル位置の設定は必ずこの方法によること。	AF
#FLGET (2021H)	カーソル位置で，カーソル点滅1文字入力を行い，Aに押されたキャラクタをセット。オートリピートもかかる(MZ-80K/C/1200は不可)。画面へのエコーバックは行わない。	AF
#RDVSW (2024H)	デフォルトデバイスをAに読み出す。デフォルトを知りたいときには必ずこの方法によるものとする。	A
#SDVSW (2027H)	デフォルトにしたいデバイス名をAに入れコールすると，デフォルトデバイスがセットされる。今後必ずこの方法によること。(#DVSW)を直接触ることも禁止する。	AF
#INP (202AH)	共通I/Oポートから1バイトをAに読み込む。ポートはCで指定する。	AF
#OUT (202DH)	共通I/OポートへAを出力する。ポートはCで指定する。	なし
#WIDCH (2030H)	画面のモード(40キャラ，80キャラ)を切り換える。Aに40以下の数をセットすると40キャラ，40より大きい数をセットしてコールすると80キャラとなる。現在のモードは(#WIDTH)に入っている。この機能はMZ-80K/C/1200/700/1500にはない。	AF BC DE HL
#ERROR (2033H)	Aにエラー番号をセットしてコールすることによりエラーメッセージを表示する。	〃

表10 S-OSのワークエリア

ワーク名 (アドレス，バイト数)	内　　容
#USR (1F7EH～，2バイト)	CIOSをコールドスタートしたあとジャンプするアドレスを示している。通常はS-OSのホットスタートのアドレスになっている。
#DVSW (1F7DH，1バイト)	テープフォーマットなどを切り換えるフラグ。 0：MZフォーマット2400ボー(共通モード) 1：各機種のモニタに依存 3：QD(MZ-1500のみ) コールドスタート時は0になっている。
#LPSW (1F7CH，1バイト)	#PRINT～#TAB，#PRTHL，#PRTHXルーチンでの出力をディスプレイだけでなくプリンタにも出力するかどうかのフラグ。0以外でプリンタにも出力。コールドスタート時は0になっている。
#PRCNT (1F7AH～，2バイト)	改行してから表示した文字数を格納してあるアドレスを示している。
#XYADR (1F78H～，2バイト)	カーソル座標が格納されているアドレスを示している。
#KBFAD (1F76H～，2バイト)	各機種のキー入力用バッファのアドレスを示している。 例) LD DE,(#KBFAD) CALL #GETL
#IBFAD (1F74H～，2バイト)	インフォメーションブロックの先頭アドレスを示している。同時にファイルアトリビュートのアドレスでもある。
#SIZE (1F72H～，2バイト)	ファイルサイズ。#WOPEN，#WRD，#FCB，#RDD，#ROPENルーチンで使用される。
#DTADR (1F70H～，2バイト)	ファイル先頭アドレス。
#EXADR (1F6EH～，2バイト)	ファイルのエントリアドレス。
#STKAD (1F6CH～，2バイト)	各機種のモニタが使用しているスタックのアドレスを示している。
#MEMAX (1F6AH～，2バイト)	S-OSで使用できるメモリの上限を表す。
#WKSIZ (1F68H～，2バイト)	特殊ワークエリアのサイズを表す。
#DIRNO (1F67H，1バイト)	#FCBで使用するワーク。このワーク値を入れて#FCBをコールすると，先頭から数えてその値で示されるFCBを(#IBFAD)にロードする。ロード後，値は1増える。
#MXTRK (1F66H，1バイト)	使用できる最大トラック数が入っている。
#DTBUF (1F64H～，2バイト)	ディスクからデータを読み込む先頭アドレスが入っている。データバッファは256バイト。
#FATBF (1F62H～，2バイト)	ディスクからFATを読み込む先頭アドレスが入っている。FATバッファは256バイト。
#DIRPS (1F60H～，2バイト)	ディレクトリが入っているレコードナンバーの始まりを示す。S-OS"SWORD"では10H，書き換えることによってディレクトリの位置を移動できる。
#FATPOS (1F5EH～，2バイト)	ファイルアロケーションテーブル(FAT)が入っているレコードナンバーを示す。S-OS"SWORD"では0EH。書き換えることによりFATの位置を移動することができる。
#DSK (1F5DH，1バイト)	アクセスしようとするデバイス名が入る。
#WIDTH (1F5CH，1バイト)	現在のスクリーンモードが入っている。 40キャラの場合 ： 28H 80キャラの場合 ： 50H MZ-80K/C/1200/700/1500は横40キャラ固定。
#MAXLN (1F5BH，1バイト)	画面に表示できる最大行数が入っている。

リスト1　"SWORD"本体(Z80カード版)

```
3800 20 27 53 4F 53 53 59 53 : 3B
3808 56 31 00 08 80 08 83 08 : A2
3810 86 08 89 08 8C 08 8F 00 : 42
3818 1B 03 1B 06 1B 09 1B 0C : 8A
3820 1B 0F 1B 12 1B 15 1B 18 : BA
3828 1B 1A 50 4F 1F 8B 10 CE : 5C
3830 06 00 30 8C D6 CE FF F2 : 57
3838 C6 0C 8D 7D 31 8C D8 CE : 3F
3840 00 00 CC C3 08 A7 C4 AE : B0
3848 A1 AF 41 33 48 5A 26 F5 : 81
3850 33 C8 36 A7 C4 AE A4 AF : 9D
3858 41 86 40 97 8B B7 FD 00 : DD
3860 4F 97 8C B7 FD 02 97 8E : 4D
3868 4C 97 8D B7 FD 03 86 14 : C1
3870 97 8A 86 FF 97 86 1C AF : 8E
3878 16 07 B8 00 00 00 00 00 : D5
-------------------------------------
SUM: 76 54 F9 70 EB 57 4C B0 : 71

3880 7E 08 92 7E 08 92 7E 08 : B6
3888 93 7E 08 92 7E 08 92 7E : 41
3890 08 92 3B 34 02 B6 FD 04 : C2
3898 84 02 26 1A 43 97 87 96 : BD
38A0 8E 2B 0C 8A 40 B7 FD 05 : 48
38A8 84 BF B7 FD 05 97 8E B6 : D7
38B0 FD 04 85 02 27 F9 35 02 : DF
38B8 3B A6 80 A7 C0 5A 26 F9 : 41
38C0 39 A6 80 A7 C0 31 3F 26 : 5C
38C8 F8 39 34 02 B6 FD 05 2B : 4A
38D0 FB 96 8E 8A 80 97 8E B7 : 05
38D8 FD 05 B6 FD 05 2A FB 35 : 14
38E0 82 34 02 B6 FC 80 8A 80 : F4
38E8 B7 FC 80 8C 34 02 96 8E : 19
38F0 84 7F B7 FD 05 97 8E 35 : 16
38F8 82 39 CE FC 82 86 0A 8D : 24
-------------------------------------
SUM: 4F 10 C2 F9 A9 16 FF E3 : BB

3900 C9 A7 C4 8D E7 8D C3 EC : E4
3908 41 DD 84 20 D4 34 46 CE : DE
3910 FC 81 CC 03 03 8D B3 6F : FE
3918 C0 ED C1 86 12 A7 C0 DC : 49
3920 84 ED C4 35 46 20 C5 34 : C9
3928 06 D6 89 CA 01 20 06 34 : 8A
3930 06 D6 89 C4 3E D7 89 86 : 4D
3938 0C 8D 8F FD FC 82 8D AC : DC
3940 35 86 34 46 CE FC 81 CC : 4C
3948 03 02 17 FF 7D 6F C0 ED : B4
3950 C1 CC 1B 39 ED C4 35 46 : 0D
3958 20 92 EC A8 10 1E 89 1F : 1C
3960 01 CE FC 82 CC 01 08 17 : 39
3968 FF 60 A7 C0 17 FF 4A 17 : 3D
3970 FF 7A 20 86 CE FC 82 86 : F1
3978 0C E6 2B C4 3F D7 89 17 : 97
-------------------------------------
SUM: 86 8C 7A A8 89 AE B9 88 : AC

3980 FF 48 ED C4 16 FF 65 EC : 5E
3988 A8 10 1E 89 1F 01 CC 0B : 56
3990 0A CE FC 82 17 FF 33 A7 : 46
3998 C0 17 FF 1D 16 FF 4D A6 : FB
39A0 2B 84 1F 97 88 39 B6 FD : D9
39A8 05 2B FB A6 2B 84 77 97 : 8E
39B0 8F 43 B7 FD 37 39 8E FD : 81
39B8 38 A6 2B E6 2B 84 07 54 : F9
39C0 54 54 54 C4 07 A7 85 39 : 2C
39C8 17 FF 64 17 FF 3F 17 FF : E5
39D0 56 8D 0D 4D 27 FB 39 17 : AF
39D8 FF 55 17 FF 30 17 FF 47 : F7
39E0 96 87 27 09 0F 87 86 1B : 84
39E8 A7 2B 16 FF 55 CE FC 82 : 88
39F0 CC 29 00 17 FE D4 ED C4 : 8F
39F8 17 FE F1 17 FE CC EC 41 : 14
-------------------------------------
SUM: 48 E3 0C 69 34 65 A2 61 : 3C

3A00 17 FE DE 5D 27 0D 81 87 : 8C
3A08 26 02 86 7B 81 F1 26 04 : C5
3A10 86 7F 21 4F A7 2B 39 A6 : 26
3A18 2A D6 87 27 06 17 FF 22 : EC
3A20 8A 40 8C 84 BF A7 2A 0F : 79
3A28 87 39 A6 2B 34 02 17 FE : DC
3A30 FE 17 FE D9 17 FE F0 0C : FD
3A38 83 CE FC 81 CC 03 01 17 : B5
3A40 FE 88 6F C0 ED C1 35 02 : 9A
3A48 81 7B 26 02 86 87 81 7F : 31
3A50 26 02 86 F1 A7 C4 81 0D : 98
3A58 26 0D 0F 83 86 11 D6 88 : BA
3A60 ED 41 86 03 B7 FC 83 17 : 04
3A68 FE 82 16 FE 8D 00 00 3F : 60
3A70 59 41 4D 41 55 43 48 49 : 51
3A78 93 D3 8F 90 9E 54 A6 84 : A1
-------------------------------------
SUM: 21 9C DA 5F 02 9A 8F BC : DD

3A80 84 BF A7 80 9C 56 25 F6 : 77
3A88 9E 5C A6 84 8A 40 A7 84 : 19
3A90 30 01 A6 84 2B 04 9C 56 : 7C
3A98 23 F6 30 1F A6 89 F8 30 : BF
3AA0 26 04 A6 84 2A F4 1F 12 : A3
3AA8 BD EA 7F BD E3 C3 7E E7 : EE
3AB0 87 17 FE 7B 17 FE 56 17 : 99
3AB8 FE 6D 0F 87 EC 2E 1E 89 : C2
3AC0 1F 01 CC 00 00 A7 80 5A : 6D
3AC8 26 FB CE FC 81 CC 04 00 : 3C
3AD0 17 FD F7 6F C0 ED C4 17 : 02
3AD8 FE 12 30 8C 90 CE FC 80 : A6

3AE0 17 FD E7 A6 43 E6 41 2A : 35
3AE8 0E 6F 41 C6 64 E7 42 17 : 28
3AF0 FD FA 17 FD D5 20 EE 4D : 3B
3AF8 27 0D 17 FD E4 4A 26 CA : 66
-------------------------------------
SUM: 80 02 6C 47 38 6B 4C E2 : 06

3B00 86 FF 97 87 17 FD C3 C6 : 40
3B08 44 17 FD AD 17 FD DD CE : C4
3B10 FC 82 86 05 17 FD B3 A7 : 77
3B18 C1 17 FD D0 EC 2E 1E 89 : 66
3B20 1F 01 D6 86 34 04 17 FD : C8
3B28 A1 E6 C0 C0 03 23 35 33 : 95
3B30 43 A6 C0 26 02 86 20 81 : F8
3B38 87 26 02 86 7B 81 F1 26 : 48
3B40 02 86 7F A7 80 6A E4 27 : A3
3B48 1B 5A 26 E5 B6 FC 81 2A : DD
3B50 13 CE FC 81 86 64 6F C0 : 77
3B58 A7 C1 17 FD 8F 17 FD 6A : 89
3B60 E6 C0 26 CD 17 FD 7A 32 : 59
3B68 61 17 FD 8E 6F 84 86 0D : 89
3B70 17 FE B9 EC 2E 1E 89 1F : AE
3B78 01 96 87 27 07 CC 1B 00 : 33
-------------------------------------
SUM: 47 3C 8A 73 EB 9F 43 74 : C1

3B80 ED 84 D7 87 39 17 FD A7 : C3
3B88 17 FD 82 17 FD 99 0F 87 : D9
3B90 CE FC 82 86 05 17 FD 32 : 1D
3B98 A7 C0 17 FD 4F 17 FD 2A : 08
3BA0 96 87 26 0A A6 C0 27 15 : EF
3BA8 17 FD 36 4A 26 E2 0F 87 : 32
3BB0 EC 2E 1E 89 1F 01 86 1B : 82
3BB8 A7 84 16 FD 3D D6 86 34 : 0B
3BC0 04 EC 2E 1E 89 1F 01 E6 : CB
3BC8 C4 33 43 C0 03 23 3D A6 : 03
3BD0 C0 26 02 86 20 81 87 26 : BC
3BD8 02 86 7B 81 F1 26 02 86 : 23
3BE0 7F A7 80 6A E4 27 1B 5A : 90
3BE8 26 E5 B6 FC 81 2A 13 CE : 49
3BF0 FC 81 86 64 6F C0 A7 C1 : FE
3BF8 17 FC F1 17 FC CC E6 C0 : 89
-------------------------------------
SUM: FB 47 1D C1 1F 1D CA 56 : 7C

3C00 26 CD 17 FC DC 32 61 6F : E4
3C08 84 16 FC EE CE FC 82 86 : 56
3C10 03 7F FC 83 20 82 17 FD : B7
3C18 16 17 FC F1 17 FD 08 EC : 22
3C20 2E 1E 89 1F 01 A6 2B 34 : FA
3C28 02 CE FC 81 CC 03 00 17 : 33
3C30 FC 98 6F C0 A7 C1 A6 80 : 51
3C38 A1 E4 26 0B F7 FC 83 17 : 43
3C40 FC AA 32 61 16 FC B3 0C : 0A
3C48 83 81 7B 26 02 86 87 81 : 35
3C50 7F 26 02 86 F1 A7 C0 5C : E1
3C58 81 0D 27 0C C1 7C 25 D6 : F9
3C60 F7 FC 83 17 FC 86 20 C1 : F0
3C68 0F 83 F7 FC 83 17 FC 7C : 97
3C70 CE FC 81 CC 03 02 17 FC : 2F
3C78 51 6F C0 ED C1 86 11 D6 : 9B
-------------------------------------
SUM: 34 29 B6 AE 59 DD B9 8E : 3E

3C80 88 ED C4 17 FC 66 20 A1 : 73
3C88 17 FC 6F DC 84 ED A8 10 : 87
3C90 39 EC A8 10 DD 84 17 FC : 51
3C98 96 17 FC 71 17 FC 2B B6 : 0E
3CA0 FC 80 27 02 86 01 A7 2A : FD
3CA8 86 80 B7 FC 80 17 FC 3C : 88
3CB0 16 FC 74 CE FC 82 86 06 : 5E
3CB8 17 FC 0F A7 C0 EC A8 10 : 2D
3CC0 ED C1 ED C4 17 FC 25 17 : AE
3CC8 FC 00 B6 FC 80 26 15 A7 : 10
3CD0 2A A6 5F 81 87 26 02 86 : E5
3CD8 7B 81 F1 26 02 86 7F A7 : C1
3CE0 2B 16 FB FD 86 01 A7 2A : 91
3CE8 16 FB F6 CE FC 81 86 07 : DF
3CF0 17 FB D7 6F C0 A7 C0 EC : 6B
3CF8 A8 10 ED C1 ED C1 96 88 : 32
-------------------------------------
SUM: AB E8 E0 49 85 11 19 6F : DA

3D00 C6 01 ED C1 A6 2B 81 7B : 42
3D08 26 02 86 87 81 7F 26 02 : 5D
3D10 86 F1 A7 C4 16 FB D5 A6 : 6E
3D18 2B 81 0D 26 04 8D 02 86 : F8
3D20 0A 81 7B 26 02 86 87 81 : BC
3D28 7F 26 02 86 F1 D6 8A 34 : B2
3D30 04 8E 3B 2F F6 FD 02 C5 : B6
3D38 08 26 0F C5 02 27 03 54 : 82
3D40 24 11 30 1F 26 EE 6A E4 : E6
3D48 26 E7 A6 2A 8A 01 A7 2A : 39
3D50 43 35 82 B7 FD 01 96 8B : D0
3D58 84 BF B7 FD 00 8A 40 B7 : 78
3D60 FD 00 97 8B A6 2A 84 FE : 71
3D68 A7 2A 4F 35 82 EC 2E 1E : 0F
3D70 89 1F 03 E6 2B 34 04 A6 : 9A
3D78 C0 A1 E4 27 04 8D 9A 24 : BB
-------------------------------------
SUM: 30 A6 CA 9C 30 03 CB AD : E7

3D80 F6 35 82 EC A8 10 1E 89 : F8
3D88 34 06 E6 2D C1 80 23 02 : B3
3D90 C6 80 8E FC 80 35 40 17 : DC
3D98 FB 30 17 FB 1C 16 FB 41 : AB
3DA0 EC A8 10 1E 89 1F 01 E6 : 51
3DA8 2D C1 80 23 02 C6 80 CE : A7
3DB0 FC 80 17 FB 15 17 FB 01 : B6
3DB8 16 FB 31 00 00 3F 59 41 : 1B
3DC0 4D 41 55 43 48 49 93 D3 : 1D
3DC8 8F 90 8E 00 00 1F 10 84 : 60
3DD0 C0 34 02 DC 1F 84 3F C4 : 78
3DD8 E0 43 53 C3 00 01 30 8B : F5
3DE0 1F 10 84 3F AA E0 1F 01 : 9C
3DE8 B5 D4 09 20 07 A6 84 B7 : 9A
3DF0 D3 83 20 04 86 00 A7 84 : 2B
3DF8 B7 D4 09 39 B6 FD 05 2B : B0
-------------------------------------
SUM: F0 52 D3 CA F9 86 B2 E6 : F6

3E00 FB 96 8F 8A 07 43 B7 FD : A8
3E08 37 EC 2E 1E 89 ED 8C BB : 2C
3E10 86 21 A7 8C D6 30 8C A3 : 0F
3E18 CE FC 80 C6 41 17 FA AA : 0C
3E20 17 FA 96 17 FA C6 17 FA : 8F
3E28 A1 B6 FC 83 A7 2B 17 FA : B9
3E30 B0 96 8F 43 B7 FD 37 39 : 3C
3E38 B6 FD 05 2B FB 96 8F 8A : 8D
3E40 07 43 B7 FD 37 EC 2E 1E : 6D
3E48 89 ED 8D FF 7E 86 20 A7 : CD
3E50 8C 99 A6 2B A7 8C 9E 30 : F7
3E58 8D FF 60 CE FC 80 C6 41 : 3D
3E60 17 FA 67 17 FA 53 17 FA : ED
3E68 83 B6 FD 05 2B FB 96 8F : 86
3E70 43 B7 FD 37 39 00 00 3F : A6
3E78 59 41 4D 41 55 43 48 49 : 51
-------------------------------------
SUM: 83 52 02 8B 05 0A 64 03 : D8

3E80 93 D3 8F 90 8E 00 00 B5 : C8
3E88 D4 09 20 07 A6 84 B7 D3 : B8
3E90 83 20 04 86 00 A7 84 B7 : 0F
3E98 D4 09 39 EC 2E 1E 89 ED : C4
3EA0 8C E3 86 21 A7 8C E3 30 : 5C
3EA8 8C CB CE FC 80 C6 26 17 : A4
3EB0 FA 18 17 FA 04 17 FA 34 : 6C
3EB8 17 FA 0F B6 FC 83 A7 2B : 27
3EC0 16 FA 1E EC 2E 1E 89 ED : DC
3EC8 8C BB 86 20 A7 8C BB A6 : 81
3ED0 2B A7 8C C0 30 8C 9E CE : 46
3ED8 FC 80 C6 26 17 F9 EB 17 : 7A
3EE0 F9 D7 16 FA 07 86 0A 8C : 03
3EE8 86 09 34 09 C6 FD 1F 9B : 49
3EF0 32 78 30 E4 A7 84 EC A8 : 7D
3EF8 10 1E 89 1F 03 ED 02 A6 : 6E
-------------------------------------
SUM: 71 17 5F CE 1C 58 52 BF : 3A

3F00 2E 84 0F 4C A7 05 A6 2E : 8D
3F08 5F 85 10 27 01 5C E7 06 : 65
3F10 EC 2E 1E 89 58 49 58 49 : 03
3F18 58 49 84 3F A7 04 A6 2C : E1
3F20 84 03 A7 07 E6 2B 34 74 : EE
3F28 8D 4B 35 74 A7 01 26 23 : 72
3F30 5A 27 30 33 C9 01 00 EF : 9D
3F38 02 6C 05 A6 05 81 10 23 : D2
3F40 E5 86 01 A7 05 6C 06 A6 : 30
3F48 06 81 01 23 D9 6F 06 6C : 65
3F50 04 20 D3 80 0A 27 06 4A : F8
3F58 27 06 86 01 8C 86 02 8C : 54
3F60 86 04 21 4F E6 2A A7 2B : DC
3F68 26 03 C4 FE 8C CA 01 E7 : 29
3F70 2A 32 68 35 89 A6 84 81 : 2D
3F78 0A 26 03 7E FE 08 7E FE : 33
-------------------------------------
SUM: 34 ED 7D DA 6F 86 B3 CB : EB

3F80 05 EC A8 10 1E 89 1F 01 : 70
3F88 6E 84 17 F9 3D 30 8D 00 : FC
3F90 34 CE FC 80 F6 00 12 A6 : 2C
3F98 80 A7 C0 5A 26 F9 17 F9 : 70
3FA0 4B 30 8D 00 23 CE FC 80 : 75
3FA8 C6 0F A6 80 A7 C0 5A 26 : E2
3FB0 F9 17 F9 38 4F B7 FD 37 : 7B
3FB8 8E FD 38 A7 80 4C 81 07 : BE
3FC0 26 F9 7E FE 00 00 00 01 : 9C
3FC8 00 00 3F 59 41 4D 41 55 : BC
3FD0 43 48 49 93 E0 00 90 F9 : D0
3FD8 22 F9 83 F9 9D F9 B0 F9 : D6
3FE0 C8 F9 CF F9 DF F9 F1 FA : 4C
3FE8 00 FA 40 FA 53 FA DA FB : 56
3FF0 AE FC 3F FC B1 FC BA FC : 48
3FF8 DC FD 14 FD 40 FD 96 FD : BA
-------------------------------------
SUM: 9C 5E CA 11 F1 75 45 BA : 3A

4000 AC FD C9 FE 25 FE 61 FE : F2
4008 C4 FE EC FF 0E FF 11 F9 : C4
4010 22 FF AA FF B3 33 8C BF : FB
4018 DC 80 1E 89 1F 02 A6 A4 : 6E
4020 D6 82 58 C1 3E 23 04 5F : 35
4028 8A 01 8C 84 FE A7 A4 EC : D0
4030 C5 AD CB 86 01 B7 FD 05 : 7D
4038 12 20 DA 00 00 00 00 00 : 0C
4040 00 00 00 00 00 00 00 00 : 00
4048 00 00 00 00 00 00 00 00 : 00
4050 00 00 00 00 00 00 00 00 : 00
4058 00 00 00 00 00 00 00 00 : 00
4060 00 00 00 00 00 00 00 00 : 00
4068 00 00 00 00 00 00 00 00 : 00
4070 00 00 00 00 00 00 00 00 : 00
4078 00 00 00 00 00 00 00 00 : 00
-------------------------------------
```

```
SUM: A5 CA 06 50 42 B3 49 AA : AD

4080 31 00 03 3A 5C 1F FE 50 : 37
4088 3E 50 C4 AA 1B 11 A2 10 : DA
4090 CD 13 1C C3 A2 16 00 04 : 7B
4098 00 00 00 00 00 00 00 00 : 00
40A0 00 00 53 57 4F 52 44 2D : BC
40A8 46 4D 37 20 5A 2D 38 30 : D9
40B0 20 44 65 62 75 67 67 65 : D3
40B8 72 0D 00 2D 2D 3E 20 42 : 79
40C0 72 65 61 6B 20 61 74 20 : B8
40C8 24 00 57 68 61 74 3F 0D : 04
40D0 00 42 72 65 61 6B 20 70 : 75
40D8 6F 69 6E 74 20 69 73 20 : D6
40E0 24 00 2A 2A 2A 2A 0D 00 : D9
40E8 41 64 64 72 20 3E 20 2B : 24
40F0 30 20 2B 31 20 2B 32 20 : 49
40F8 2B 33 20 2B 34 20 2B 35 : 5D
---------------------------------
SUM: D9 C8 43 51 04 C6 73 A5 : 17

4100 20 2B 36 20 2B 37 20 2B : 4E
4108 38 20 2B 39 20 2B 41 20 : 68
4110 2B 42 20 2B 43 20 2B 44 : 8A
4118 20 2B 45 20 2B 46 20 3A : 7B
4120 20 53 75 6D 20 20 30 31 : F6
4128 32 33 34 35 36 37 38 39 : AC
4130 41 42 43 44 45 46 0D 00 : A2
4138 20 3E 20 00 3A 20 00 20 : F8
4140 20 20 00 23 23 23 23 23 : EF
4148 23 23 23 23 23 23 23 23 : 18
4150 23 23 23 23 23 23 23 23 : 18
4158 23 23 23 23 23 23 23 23 : 18
4160 23 23 23 23 23 23 23 23 : 18
4168 23 23 23 23 23 23 23 23 : 18
4170 23 23 23 23 23 23 23 23 : 18
4178 23 23 23 0D 23 20 20 20 : F9
---------------------------------
SUM: 6B D3 C7 8C A6 9A 36 68 : 6F

4180 20 20 20 53 2D 4F 53 20 : A2
4188 53 57 4F 52 44 2F 46 4D : 51
4190 2D 37 20 5A 2D 38 30 20 : 93
4198 44 65 62 75 67 67 65 72 : 25
41A0 20 63 6F 6D 6D 61 6E 64 : FF
41A8 20 6D 65 6E 75 20 20 20 : 35
41B0 20 20 20 23 0D 23 23 23 : F9
41B8 23 23 23 23 23 23 23 23 : 18
41C0 23 23 23 23 23 23 23 23 : 18
41C8 23 23 23 23 23 23 23 23 : 18
41D0 23 23 23 23 23 23 23 23 : 18
41D8 23 23 23 23 23 23 23 23 : 18
41E0 23 23 23 23 23 23 23 23 : 18
41E8 23 23 23 23 23 0D 23 20 : FF
41F0 42 20 20 20 20 20 20 20 : 22
41F8 20 20 20 20 20 20 20 20 : 00
---------------------------------
SUM: 9B 38 1A A7 29 E0 14 D8 : 89

4200 20 23 20 44 69 73 70 6C : 5F
4208 61 79 20 63 75 72 72 65 : 1B
4210 6E 74 20 62 72 65 61 6B : 07
4218 20 70 6F 69 6E 74 2E 20 : 98
4220 20 20 20 20 20 23 0D 23 : F3
4228 20 42 20 2A 20 20 20 20 : 2C
4230 20 20 20 20 20 20 20 20 : 00
4238 20 20 23 20 52 65 6C 65 : 0B
4240 61 73 65 20 63 75 72 72 : 15
4248 65 6E 74 20 62 72 65 61 : 01
4250 6B 20 70 6F 69 6E 74 2E : E3
4258 20 20 20 20 20 20 23 0D : F0
4260 23 20 42 20 61 64 72 31 : 0D
4268 20 20 20 20 20 20 20 20 : 00
4270 20 20 20 23 20 53 65 74 : CF
4278 20 62 72 65 61 6B 20 70 : B5
---------------------------------
SUM: 63 05 AF 93 C0 3D AF 67 : BD

4280 6F 69 6E 74 20 6F 6E 20 : D7
4288 28 61 64 72 31 29 2E 20 : 07
4290 20 20 20 20 20 20 20 23 : 03
4298 0D 23 20 43 20 61 64 72 : EA
42A0 31 2C 61 64 72 32 20 20 : 06
42A8 20 20 20 20 20 23 20 43 61 : 67
42B0 6C 69 63 75 6C 61 74 65 : 53
42B8 20 61 64 72 31 2B 61 64 : 78
42C0 72 32 2C 61 64 72 31 2D : 65
42C8 61 64 72 32 2E 20 20 20 : F7
42D0 23 0D 23 20 44 5B 61 64 : D7
42D8 72 31 5B 2C 61 64 72 32 : 93
42E0 5D 5D 20 20 20 23 20 44 : A1
42E8 75 6D 70 20 66 72 6F 6D : 26
42F0 20 61 64 72 31 20 74 6F : 8B
42F8 20 61 64 72 32 2E 20 20 : F7
---------------------------------
SUM: 1B 83 CE B7 E3 2B 9F 42 : 12

4300 20 20 20 20 20 20 20 20 : 00
4308 20 23 0D 23 20 46 20 61 : 5A
4310 64 72 31 2C 61 64 72 32 : 9C
4318 2C 64 74 20 20 20 23 20 : A7
4320 46 69 6C 6C 20 66 72 6F : EE
4328 6D 20 61 64 72 31 20 74 : 89
4330 6F 20 61 64 72 32 20 77 : 8F
4338 69 74 68 20 64 74 2E 20 : 8B
4340 20 20 23 0D 23 20 47 5B : 55
4348 61 64 72 31 5D 20 20 20 : 25
4350 20 20 20 20 20 20 20 23 : 03
4358 20 47 6F 20 6F 62 6A 65 : 96
4360 63 74 20 70 72 6F 67 72 : 21
4368 61 6D 2E 20 20 20 20 20 : 9C
4370 20 20 20 20 20 20 20 20 : 00
4378 20 20 20 23 0D 23 20 48 : 1B
---------------------------------
SUM: 20 42 1A 34 F7 BB 6D 4A : 19

4380 20 20 20 20 20 20 20 20 : 00
4388 20 20 20 20 20 20 20 20 : 00
4390 23 20 44 69 73 70 6C 61 : A0
4398 79 20 68 65 6C 70 20 6D : CF
43A0 65 73 73 61 67 65 2E 20 : C6
43A8 20 20 20 20 20 20 20 20 : 00
43B0 20 20 20 20 23 0D 23 20 : F3
43B8 49 20 61 64 72 31 2C 6C : 69
43C0 65 74 74 65 72 20 20 20 : 84
43C8 20 23 20 49 6E 73 65 72 : 64
43D0 74 20 6D 65 6D 6F 72 79 : 2D
43D8 20 73 74 61 6D 70 20 6F : D4
43E0 6E 20 28 61 64 72 31 29 : 47
43E8 2D 2E 20 20 20 23 0D 23 : 0E
43F0 20 4A 20 61 64 72 31 20 : 12
43F8 20 20 20 20 20 20 20 20 : 00
---------------------------------
SUM: BE 35 FD 89 FD 7C 0F E0 : E1

4400 20 20 23 20 4A 75 6D 70 : 1F
4408 20 74 6F 20 61 64 72 31 : 8B
4410 2E 20 20 20 20 20 20 20 : 0E
4418 20 20 20 20 20 20 20 20 : 00
4420 20 20 20 20 20 20 23 0D : F0
4428 23 20 4D 5B 61 64 72 31 : 53
4430 5D 20 20 20 20 20 20 20 : 3D
4438 20 20 20 20 45 78 63 : C3
4440 68 61 6E 67 65 20 63 6F : F5
4448 6E 74 65 6E 74 73 20 6F : 2B
4450 66 20 6D 65 6D 6F 72 79 : 1F
4458 2E 20 20 20 20 20 20 23 : 11
4460 0D 23 20 50 20 20 20 20 : 20
4468 20 20 20 20 20 20 20 20 : 00
4470 20 20 20 20 23 20 43 6F : 75
4478 6D 70 6C 65 6D 65 6E 74 : 62
---------------------------------
SUM: 72 3C AB 8D E2 E9 52 3F : 42

4480 20 70 72 69 6E 74 65 72 : 24
4488 20 73 77 69 74 63 68 2E : E0
4490 20 20 20 20 20 20 20 20 : 00
4498 23 0D 23 20 51 20 20 20 : 24
44A0 20 20 20 20 20 20 20 20 : 00
44A8 20 20 20 20 20 23 20 51 : 34
44B0 75 69 74 20 74 6F 20 53 : C8
44B8 57 4F 52 44 2E 20 20 20 : CA
44C0 20 20 20 20 20 20 20 20 : 00
44C8 20 20 20 20 20 20 20 20 : 00
44D0 20 23 0D 23 20 52 20 20 : 25
44D8 20 20 20 20 20 20 20 20 : 00
44E0 20 20 20 20 20 20 23 20 : 03
44E8 44 69 73 70 6C 61 79 2F : 05
44F0 65 78 63 68 61 6E 67 65 : 43
44F8 20 72 65 67 69 73 74 65 : 13
---------------------------------
SUM: F8 FE FA 98 0B FD 84 5D : 71

4500 72 73 2E 20 20 20 20 20 : B3
4508 20 20 23 0D 23 20 53 20 : 26
4510 61 64 72 31 2C 61 64 72 : CB
4518 32 2C 64 74 20 20 20 23 : B9
4520 20 53 65 72 63 68 20 22 : 57
4528 64 74 22 20 66 72 6F 6D : CE
4530 20 61 64 72 31 20 74 6F : 8B
4538 20 61 64 72 32 2E 20 20 : F7
4540 20 20 20 23 0D 23 20 54 : 27
4548 20 61 64 72 31 2C 61 64 : 79
4550 72 32 2C 61 64 72 33 20 : 5A
4558 23 20 54 72 61 6E 73 66 : B1
4560 65 72 20 66 72 6F 6D 20 : CB
4568 61 64 72 31 2D 61 64 72 : CC
4570 32 20 74 6F 20 61 64 72 : 8C
4578 33 2E 20 20 23 0D 23 23 : 17
---------------------------------
SUM: E9 A3 A0 D6 A0 56 99 58 : E9

4580 23 23 23 23 23 23 23 23 : 18
4588 23 23 23 23 23 23 23 23 : 18
4590 23 23 23 23 23 23 23 23 : 18
4598 23 23 23 23 23 23 23 23 : 18
45A0 23 23 23 23 23 23 23 23 : 18
45A8 23 23 23 23 23 23 23 23 : 18
45B0 23 23 23 23 23 23 0D 00 : DF
45B8 50 72 69 6E 74 65 72 20 : 04
45C0 69 73 20 65 6E 61 62 6C : FE
45C8 65 2E 0D 00 50 72 69 6E : 39
45D0 74 65 72 20 69 73 20 64 : CB
45D8 69 73 65 6E 61 62 6C 65 : 43
45E0 2E 0D 00 20 20 20 20 20 : DB
45E8 41 46 20 20 20 42 43 20 : 8C
45F0 20 20 44 45 20 20 20 48 : 71
45F8 4C 20 20 20 41 46 27 20 : 7A
---------------------------------
SUM: CB 73 E6 FB 92 CA 52 3D : 0A

4600 20 42 43 27 20 20 44 45 : 95
4608 27 20 20 48 4C 27 20 20 : 62
4610 49 58 20 20 20 49 59 20 : C3
4618 20 20 53 50 20 20 20 50 : 93
4620 43 0D 2D 52 3E 20 00 F3 : 20
4628 16 FB 16 4E 17 A4 17 F3 : 3A
4630 16 3B 18 72 18 A1 18 A7 : 53
4638 18 C0 18 F3 16 F3 16 CB : CD
4640 18 F3 16 F3 16 1E 19 35 : 96
4648 19 38 19 93 19 DE 19 F3 : 00
4650 16 F3 16 F3 16 F3 16 F3 : 24
4658 16 F3 16 13 1A FE 01 D8 : 23
4660 FE 21 38 F7 C9 E6 DF FE : DA
4668 41 D8 FE 5B 3F C9 F5 7E : ED
4670 23 66 6F F1 C9 F1 ED 73 : 03
4678 96 10 E3 22 98 10 E3 31 : 67
---------------------------------
SUM: 8C 5D 2C D5 F7 A5 0F 40 : D5

4680 14 03 FD E5 DD E5 08 D9 : 9C
4688 E5 D5 C5 F5 08 D9 E5 D5 : 0F
4690 C5 F5 11 BB 10 CD 13 1C : 92
4698 2A 98 10 2B CD E4 1B CD : 96
46A0 C8 1B 31 00 03 21 A2 16 : F0
46A8 E5 3E 2D CD F5 1B 11 00 : 3E
46B0 01 CD 37 1C CD 5C 16 D8 : 38
46B8 FE 1B C8 FE 2D 20 17 CD : 10
46C0 5B 16 D8 CD 65 16 38 2B : F4
46C8 21 27 16 D6 41 17 4F 06 : E1
46D0 00 09 CD 6E 16 E9 CD 73 : 83
46D8 1C 38 18 CD 5C 16 FE 3E : E7
46E0 20 11 13 CD 5C 16 D8 FE : 59
46E8 3A C8 CD 7C 1C 38 04 77 : 1A
46F0 23 18 F0 11 CA 10 CD 13 : F6
46F8 1C 18 A7 2A 9F 10 CD 5B : DC
---------------------------------
SUM: C5 2D 8A 09 AD C1 C3 17 : CD

4700 16 30 18 11 D1 10 CD 13 : 30
4708 1C 3A 9E 10 B7 20 06 11 : F2
4710 E2 10 C3 13 1C CD E4 1B : B0
4718 C3 C8 1B FE 2A 20 0F 3A : 37
4720 9E 10 B7 C8 3A A1 10 77 : 8F
4728 AF 32 9E 10 18 D5 44 4D : 0D
4730 CD 73 1C 38 BE 3A 9E 10 : 3A
4738 B7 28 04 3A A1 10 02 7E : 4E
4740 32 A1 10 3E FF 77 32 9E : 67
4748 10 22 9F 10 18 B5 CD 5B : D6
4750 16 CD 73 1C 38 9D CD 5C : 70
4758 16 38 98 FE 2C 20 94 44 : 08
4760 4D CD 5B 16 CD 73 1C 38 : 1F
4768 8A 50 59 EB CD E4 1B E5 : CF
4770 D5 3E 2B CD F5 1B EB CD : D3
4778 E4 1B 3E 3D CD F5 1B 00 : 57
---------------------------------
SUM: A6 5D E0 EF 56 2D 57 4E : FA

4780 19 CD E4 1B D1 E1 CD C8 : 2C
4788 1B CD E4 1B 3E 2D CD F5 : 14
4790 1B EB CD E4 1B 3E 3D CD : 1A
4798 F5 1B EB B7 ED 52 CD E4 : A2
47A0 1B C3 C8 1B 2A 9A 10 CD : 62
47A8 5B 16 38 09 CD 73 1C DA : E8
47B0 F3 16 22 9A 10 24 2E 00 : 27
47B8 CD 5C 16 38 12 FE 2C C2 : 75
47C0 F3 16 CD 5B 16 CD 73 1C : A3
47C8 DA F3 16 11 10 00 19 3E : 5B
47D0 F0 A5 6F 22 9C 10 2A 9A : 96
47D8 10 3E F0 A5 6F 11 E8 10 : 5B
47E0 CD 13 1C 01 00 10 CD E4 : BE
47E8 1B 11 38 11 CD 13 1C 7E : EF
47F0 81 4F 7E 23 CD E9 1B CD : 0F
47F8 DF 1B 10 F3 11 3C 11 CD : 28
---------------------------------
SUM: 8F 65 DC 22 0C 03 DD D7 : B5

4800 13 1C 79 CD E9 1B 11 3F : C9
4808 11 CD 13 1C 11 F0 FF 19 : 26
4810 06 10 7E FE 21 30 02 3E : 23
4818 20 CD F5 1B 23 10 F3 CD : F0
4820 C8 1B 22 9A 10 CD 3A 1C : D2
4828 C8 ED 5B 9C 10 B7 ED 52 : B2
4830 19 C8 7D B7 20 AD CD C8 : 77
4838 1B 18 A2 CD 5B 16 CD 73 : 53
4840 1C DA F3 16 CD 5C 16 FE : 3C
4848 2C C2 F3 16 44 4D CD 5B : B0
4850 16 CD 73 1C DA F3 16 CD : 22
4858 5C 16 FE 2C C2 F3 16 CD : 34
4860 5B 16 CD 7C 1C DA F3 16 : B9
4868 23 02 03 B7 ED 42 09 20 : 37
4870 F8 C9 CD 5B 16 38 0E CD : 12
4878 73 1C DA F3 16 EB 00 2A : 87
---------------------------------
SUM: B1 2A 69 B1 BB 60 DF 2C : 1B

4880 96 10 73 23 72 31 00 03 : E2
4888 F1 C1 D1 E1 08 D9 F1 C1 : F7
4890 D1 E1 08 D9 DD E5 FD E5 : 37
4898 ED 7B 96 10 E3 2B E3 FB : FA
48A0 C9 11 43 11 C3 13 1C CD : ED
48A8 5B 16 CD 73 1C DA F3 16 : B0
48B0 CD 5C 16 FE 2C C2 F3 16 : 34
48B8 13 1A B7 C8 77 23 18 F8 : 56
48C0 CD 5B 16 CD 73 1C DA F3 : 67
48C8 16 FB E9 2A 9A 10 CD 5B : F6
48D0 16 38 06 CD 73 1C DA F3 : 7D
48D8 16 CD E4 1B 3E 3A CD F5 : 1C
48E0 1B 7E CD E9 1B 3E 2D CD : A2
48E8 F5 1B 11 00 01 CD 37 1C : 42
48F0 1A B7 28 27 FE 1B C8 CD : CE
48F8 5C 16 CD 73 1C DA F3 16 : B1
---------------------------------
SUM: DE 8B 7B 99 B0 6E 58 97 : 8A

4900 1A FE 3A C2 F3 16 13 13 : 43
4908 13 1A FE 2D C2 F3 16 CD : F0
4910 5B 16 38 07 CD 7C 1C DA : EF
4918 F3 16 77 23 18 BB 3A 7C : 2C
4920 1F B7 20 01 3E AF 32 7C : 92
4928 1F 11 B8 15 B7 20 03 11 : E8
4930 CC 15 C3 13 1C C3 29 1B : DA
4938 21 00 03 CD 5B 16 30 2A : BC
```

▶ 6月号の「マシン語はカラムーチョより辛いか」に登場した丸貧じいさんが好きです。
岡部　孝弘 (17) 埼玉県

```
4940 11 E3 15 CD 13 1C 06 0A : 15
4948 E5 CD 6E 16 CD E4 1B E1 : E3
4950 CD DF 1B 23 23 10 F1 2A : 38
4958 96 10 CD E4 1B CD DF 1B : 39
4960 CD 6E 16 2B CD E4 1B C3 : 0B
4968 C8 1B FE 3E C2 F3 16 06 : F0
4970 0A CD 5B 16 D8 E5 CD 73 : 45
4978 1C DA F3 16 4D 7C E1 71 : 1A
---------------------------------
SUM: BA F0 52 8E D8 FD DD E5 : 21

4980 23 77 23 10 EC CD 5B 16 : F7
4988 D8 CD 73 1C DA F3 16 22 : 39
4990 96 10 C9 CD 5B 16 CD 73 : ED
4998 1C DA F3 16 44 4D CD 5C : B9
49A0 16 FE 2C C2 F3 16 CD 5B : 33
49A8 16 CD 73 1C DA F3 16 CD : 22
49B0 5C 16 FE 2C C2 F3 16 CD : 34
49B8 5B 16 CD 7C 1C DA F3 16 : B9
49C0 50 59 EB 13 BE 20 08 08 : 95
49C8 CD E4 1B CD DF 1B 08 23 : BE
49D0 CD 3A 1C 28 06 B7 ED 52 : 47
49D8 19 20 E9 C3 C8 1B CD 5B : F0
49E0 16 CD 73 1C DA F3 16 CD : 22
49E8 5C 16 FE 2C C2 F3 16 22 : 89
49F0 9A 10 CD 5B 16 CD 73 1C : 44
49F8 DA F3 16 CD 5C 16 FE 2C : 4C
---------------------------------
SUM: 79 A2 1B D0 89 CF 5E 21 : DD

4A00 C2 F3 16 22 9C 10 CD 5B : C1
4A08 16 CD 73 1C DA F3 16 ED : 42
4A10 5B 9C 10 13 B7 ED 52 19 : 29
4A18 30 23 ED 5B 9A 10 B7 ED : E9
4A20 52 19 C8 38 18 EB 2A 9C : 34
4A28 10 ED 4B 9A 10 B7 ED 42 : D8
4A30 44 4D 62 6B 09 EB 2A 9C : 18
4A38 10 03 ED B8 C9 54 5D 2A : 5C
4A40 9C 10 ED 4B 9A 10 B7 ED : 32
4A48 42 44 4D 2A 9A 10 03 ED : 97
4A50 B0 C9 00 00 00 00 00 00 : 79
4A58 00 00 00 00 00 00 00 00 : 00
4A60 00 00 00 00 00 00 00 00 : 00
4A68 00 00 00 00 00 00 00 00 : 00
4A70 00 00 00 00 00 00 00 00 : 00
4A78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: A7 F2 22 16 F5 01 44 CC : D7

4A80 00 00 00 00 00 00 00 00 : 00
4A88 00 00 00 00 00 00 00 00 : 00
4A90 00 00 00 00 00 00 00 00 : 00
4A98 00 00 00 00 00 00 00 00 : 00
4AA0 00 00 00 00 00 00 00 00 : 00
4AA8 00 00 00 00 00 00 00 00 : 00
4AB0 00 00 00 00 00 00 00 00 : 00
4AB8 00 00 00 00 00 00 00 00 : 00
4AC0 00 00 00 00 00 00 00 00 : 00
4AC8 00 00 00 00 00 00 00 00 : 00
4AD0 00 00 00 00 00 00 00 00 : 00
4AD8 00 00 00 00 00 00 00 00 : 00
4AE0 00 00 00 00 00 00 00 00 : 00
4AE8 00 00 00 00 00 00 00 00 : 00
4AF0 00 00 00 00 00 00 00 00 : 00
4AF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 00 00 00 00 00 00 00 00 : 00

4B00 C3 29 1B C3 0C 1E C3 1B : D2
4B08 1B C3 1B 1B C3 1B 1B C3 : D0
4B10 1B 1B C3 1B 1B C3 1C 1B : 29
4B18 C3 1B 1B C9 F5 ED 57 EA : E5
4B20 75 16 CF 00 F1 FB C9 ED : FC
4B28 45 F3 ED 56 31 00 04 3E : EE
4B30 50 CD AA 1B 3E 3F CF 02 : 30
4B38 21 8C 1B CF 03 3E 07 CF : AE
4B40 04 3E 44 CF 05 21 96 1B : 2C
4B48 06 08 7E 23 CF 06 10 FA : 8E
4B50 21 00 80 22 68 1F AF 32 : 2B
4B58 7C 1F 32 7D 1F 21 00 16 : A0
4B60 CF 0F CD FF 1B 0C 3C 3C : 49
4B68 3C 3C 3C 20 53 2D 4F 53 : F6
4B70 20 53 57 4F 52 44 20 3E : 0D
4B78 3E 3E 3E 3E 0D 00 3E 01 : 44
---------------------------------
SUM: F7 C5 A7 3F 6A 45 32 0A : 8D

4B80 32 A8 1B AF 32 5F 1B FB : 4B
4B88 2A 7E 1F E9 80 80 80 80 : B0
4B90 80 80 80 80 80 80 00 10 : 10
```

```
4B98 20 30 47 57 67 77 21 20 : 0D
4BA0 40 C9 00 50 19 00 19 00 : 8B
4BA8 00 01 21 A3 1B FE 29 3E : 45
4BB0 28 38 01 17 32 5C 1F 77 : 9C
4BB8 2B CF 01 B7 C9 F5 E5 CF : 24
4BC0 0E 7D E1 B7 20 03 F1 C9 : 00
4BC8 F5 3E 0D 18 29 F5 3E 07 : BB
4BD0 18 24 3A 83 00 90 3F D8 : A0
4BD8 CD DF 1B 3C 20 FA C9 F5 : DB
4BE0 3E 20 18 12 7C CD E9 1B : D5
4BE8 7D F5 0F 0F 0F 0F CD F2 : 6D
4BF0 1B F1 CD 6B 1C F5 CF 0A : 2E
4BF8 CD 1E 1C CF 12 F1 C9 E3 : 85
---------------------------------
SUM: 1A 89 77 19 EA 69 87 C6 : D3

4C00 EB CD 13 1C EB F5 7E 23 : 68
4C08 B7 20 FB F1 E3 C9 F5 3E : A2
4C10 0D 18 02 F5 AF CF 0D CD : 74
4C18 1E 1C CF 13 F1 C9 E5 2A : E5
4C20 7C 1F 2C 2D E1 C0 F1 F1 : 77
4C28 C9 CF 12 C9 F5 18 02 F5 : 77
4C30 3E AF 32 7C 1F F1 C9 CF : 43
4C38 0B C9 CF 09 C9 CF 08 C9 : 15
4C40 CF 07 C9 CF 0E C9 CF 10 : 24
4C48 C9 CF 0F C9 CF 08 B7 28 : 26
4C50 0E FE 1B 28 0F FE 20 20 : 9C
4C58 06 CF 07 FE 1B 28 05 E3 : 05
4C60 23 23 E3 C9 E3 7E 23 66 : DC
4C68 6F E3 C9 F6 F0 27 C6 A0 : 8E
4C70 CE 40 C9 CD 7C 1C 67 D4 : 77
4C78 7C 1C 6F C9 C5 1A 13 CD : 8F
---------------------------------
SUM: E3 8C FC A3 47 C0 37 B8 : 04

4C80 93 1C 38 0D 0F 0F 0F 0F : 30
4C88 4F 1A 13 CD 93 1C 38 01 : 31
4C90 B1 C1 C9 D6 30 D8 FE 0A : 21
4C98 38 07 FE 11 D8 D6 07 FE : 01
4CA0 10 3F C9 EB CF 16 EB C9 : 9C
4CA8 EB CF 17 EB C9 CF 16 13 : 7D
4CB0 77 23 0B 78 B1 30 F6 C9 : BD
4CB8 7E 23 CF 17 13 0B 78 B1 : CE
4CC0 20 F6 C9 C5 06 FD 1A C1 : 82
4CC8 17 CB 1F C9 C5 06 FD 02 : 94
4CD0 C1 C9 CF 1E CD E9 1C D5 : 1E
4CD8 21 FA 1D 11 00 07 01 12 : 63
4CE0 00 ED B0 D1 CD C1 1D B7 : D0
4CE8 C9 21 FA 1D 77 23 32 1F : EC
4CF0 29 CD 4B 1D CD 15 29 D8 : 41
4CF8 32 5D 1F 06 0D CD 37 1D : E2
---------------------------------
SUM: F8 0E B4 F4 BC B2 9E E3 : 9D

4D00 CD 8D 1D 20 01 1B 77 13 : 3D
4D08 23 10 F2 1A FE 2E 20 01 : 8C
4D10 13 06 03 CD 37 1D 77 13 : C7
4D18 23 10 F8 36 20 3A 5D 1F : 37
4D20 CD 18 29 C0 FE 53 C8 21 : 08
4D28 0B 1E 06 11 7E FE 21 D0 : AD
4D30 3E 0D 77 2B 10 F6 C9 D5 : 91
4D38 CD C1 1D 1A D1 FE 3A 28 : F6
4D40 06 FE 20 38 02 1A C9 3E : 7F
4D48 20 1B C9 CD C1 1D 13 1A : DC
4D50 1B FE 3A C2 AD 25 1A 13 : 14
4D58 13 FE 61 D8 FE 7B D0 D6 : 69
4D60 20 C9 11 01 07 06 0D 1A : 2F
4D68 CD 41 1D CD 8D 1D CD F5 : 64
4D70 1B 13 10 F3 3E 2E CD F5 : 5F
4D78 1B 06 03 1A CD 41 1D CD : 36
---------------------------------
SUM: 80 EF 92 CD C0 4E E1 46 : 03

4D80 F5 1B 13 10 F6 CF 08 FE : FE
4D88 20 C0 CF 07 C9 FE 2E C0 : 6B
4D90 3E 20 C9 E6 87 47 21 00 : FC
4D98 07 7E E6 87 B8 20 1D 3A : 21
4DA0 20 29 F5 3A 5D 1F 32 20 : 46
4DA8 29 CD E9 1C F1 32 20 29 : 67
4DB0 11 00 07 21 FA 1D 06 10 : 66
4DB8 CD C7 1D C0 3E 08 B7 C9 : 37
4DC0 13 1A FE 20 28 FA C9 13 : 49
4DC8 23 7E FE 21 30 02 AF C9 : 6A
4DD0 7E CD 8D 1D 4F 1A CD 8D : B8
4DD8 1D B9 C0 FE 0D C8 23 13 : 9F
4DE0 10 EE AF C9 C3 F6 1D C3 : 0F
4DE8 F6 1D C3 F6 1D C3 F6 1D : BF
4DF0 C3 F6 1D C3 F6 1D 3E 02 : EC
4DF8 37 C9 00 00 00 00 00 00 : 00
---------------------------------
```

```
SUM: 52 1E 6B 99 0E 5E 3C 78 : 94

4E00 00 00 00 00 00 00 00 00 : 00
4E08 00 00 00 00 E3 F5 7E 23 : 79
4E10 32 82 00 F1 E3 FD E5 DD : 47
4E18 E5 E5 D5 C5 F5 08 D9 E5 : 1F
4E20 D5 C5 F5 F5 ED 73 80 00 : 64
4E28 AF 32 05 FD 00 F1 38 0F : 1B
4E30 F1 C1 D1 E1 08 D9 F1 C1 : F7
4E38 D1 E1 DD E1 FD E1 C9 11 : 28
4E40 52 1E CD 13 1C 21 14 00 : A1
4E48 39 F9 E1 2B CD E4 1B C3 : CD
4E50 80 10 0D 49 6C 6C 69 67 : 8E
4E58 61 6C 20 73 79 73 74 65 : 25
4E60 6D 20 63 61 6C 6C 20 64 : AD
4E68 65 74 65 63 74 64 65 64 : 42
4E70 20 61 74 20 24 00 00 00 : 39
4E78 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: BB 88 94 48 7F CC 3F 1D : C6

4E80 00 00 00 00 00 00 00 00 : 00
4E88 00 00 00 00 00 00 00 00 : 00
4E90 00 00 00 00 00 00 00 00 : 00
4E98 00 00 00 00 00 00 00 00 : 00
4EA0 00 00 00 00 00 00 00 00 : 00
4EA8 00 00 00 00 00 00 00 00 : 00
4EB0 00 00 00 00 00 00 00 00 : 00
4EB8 00 00 00 00 00 00 00 00 : 00
4EC0 00 00 00 00 00 00 00 00 : 00
4EC8 00 00 00 00 00 00 00 00 : 00
4ED0 00 00 00 00 00 00 00 00 : 00
4ED8 00 00 00 00 00 00 00 00 : 00
4EE0 00 00 00 00 00 00 00 00 : 00
4EE8 00 00 00 00 00 00 00 00 : 00
4EF0 00 00 00 00 00 00 00 00 : 00
4EF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 00 00 00 00 00 00 00 00 : 00

4F00 00 00 00 00 00 00 00 00 : 00
4F08 00 00 00 00 00 00 00 00 : 00
4F10 00 00 00 00 00 00 00 00 : 00
4F18 00 00 00 00 00 00 00 00 : 00
4F20 00 00 00 00 00 00 00 00 : 00
4F28 00 00 00 00 00 00 00 00 : 00
4F30 00 00 00 00 00 00 00 00 : 00
4F38 00 00 00 00 00 00 00 00 : 00
4F40 00 00 00 00 00 00 00 00 : 00
4F48 00 00 00 00 00 00 00 00 : 00
4F50 00 00 00 00 00 00 00 00 : 00
4F58 00 00 00 19 50 41 0E 00 : B8
4F60 10 00 00 2E 00 2F 50 00 : BD
4F68 00 80 FF FB 00 04 00 00 : 7E
4F70 00 00 00 00 00 07 00 06 : 0D
4F78 84 00 83 00 00 00 00 21 : 28
---------------------------------
SUM: 94 80 82 42 50 7B 5E 27 : 28

4F80 E1 E9 00 00 00 00 00 00 : CA
4F88 00 00 00 00 00 00 C3 80 : 43
4F90 10 00 B3 1C C3 A3 1C 00 : 61
4F98 BE 1C C3 A8 1C C3 62 1D : A3
4FA0 C3 93 1D C3 D4 1C C3 4F : 38
4FA8 23 C3 7C 23 C3 2D 23 C3 : 5B
4FB0 B3 22 C3 73 1C C3 7C 1C : 82
4FB8 C3 93 1C C3 6B 1C C3 E4 : 63
4FC0 1B C3 E9 1B C3 CD 1B C3 : 50
4FC8 4C 1C C3 40 1C C3 3A 1C : A0
4FD0 C3 3D 1C C3 37 1C C3 2C : 21
4FD8 1C C3 2F 1C C3 29 1C C3 : F5
4FE0 D2 1B C3 FF 1B C3 13 1C : BC
4FE8 C3 0E 1C C3 BD 1B C3 C8 : 13
4FF0 1B C3 DF 1B C3 F5 1B C3 : 6E
4FF8 9E 1B C3 00 21 C3 29 1B : A4
---------------------------------
SUM: 9F F6 66 F7 92 F9 B4 3F : 70

5000 C3 44 25 C3 5A 25 C3 19 : 4A
5008 24 C3 FA 22 C3 08 25 C3 : B6
5010 26 25 C3 AC 24 C3 77 24 : 3C
5018 C3 43 1C C3 46 1C C3 49 : 53
5020 1C C3 40 1C C3 AD 25 C3 : 93
5028 C9 25 C3 C3 1C C3 CC 1C : 3B
5030 C3 AA 1B C3 6C 28 C3 D2 : 74
5038 1C                      : 1C
---------------------------------
SUM: 94 01 1C F6 D2 A4 D6 FA : ED
```

リスト2 エミュレータ

```
4080 00 00 FF 00 00 00 00 00 : FF
4088 00 00 00 00 00 00 00 00 : 00
4090 00 00 00 00 00 00 00 00 : 00
4098 00 04 00 00 00 D6 92 27 : 93
40A0 0B DE 9B E6 C0 DF 9B DE : 82
40A8 87 33 C5 39 DE 87 39 D6 : 2C
40B0 92 27 F9 2B 0B DE 9B E6 : 47
40B8 C0 DF 9B DE 95 33 C5 39 : DE
40C0 DE 9B E6 C0 DF 9B DE 97 : 0E
40C8 33 C5 39 9E 9B A6 80 9F : 2F
40D0 9B 97 80 84 07 97 82 96 : EC
40D8 80 44 44 44 84 07 97 81 : EF
40E0 96 80 5F 48 59 48 59 58 : 0F
40E8 39 CC 1E 2D FD 10 9B 20 : 18
```

```
40F0 06 CC 00 00 FD 10 9B 34 : AE
40F8 08 86 10 1F 8B 10 8E 10 : F6
---------------------------------
SUM: ED F4 63 E2 21 A4 5A 03 : 48

4100 83 0F 92 9D CB 8E 11 0E : 39
4108 AE 85 AD 84 20 F5 11 16 : A0
4110 14 40 14 65 15 43 96 80 : 3B
4118 84 0F 48 8E 11 22 AE 86 : D0
4120 6E 84 11 42 11 99 11 D8 : D8
4128 12 34 12 66 12 97 12 CC : 45
4130 12 E6 11 42 11 B7 12 05 : 2A
4138 12 4C 12 66 12 97 12 CC : 5D
4140 12 E6 96 81 48 8E 11 4C : 42
```

```
4148 AE 86 6E 84 11 5C 11 90 : 34
4150 11 5D 11 61 11 6A 11 72 : DE
4158 11 7A 11 82 39 0A 83 27 : 0B
4160 27 9E 9B A6 80 30 86 9F : DB
4168 9B 39 96 8B 85 40 27 F1 : D2
4170 20 16 96 8B 85 40 26 E9 : 2B
4178 20 0E 96 8B 85 01 27 E1 : DD
---------------------------------
SUM: 51 0B 64 93 09 75 5D 6E : 9C

4180 20 06 96 8B 85 01 26 D9 : CC
4188 DC 9B C3 00 01 DD 9B 39 : EC
4190 DC 8A 9E 93 DD 93 9F 8A : 30
4198 39 96 81 81 06 27 0D 33 : 3E
```

▶表紙が美しいので買ってしまいました。あまりよくわかりませんが，マシン語はスゴイ。
山下 正子 (29) 愛知県

```
41A0 A6 9E 9B E6 80 A6 80 9F : 0A
41A8 9B ED C4 39 9E 9B EC 81 : 2B
41B0 9F 9B D7 99 97 9A 39 96 : AA
41B8 81 84 06 81 06 27 11 EC : B6
41C0 A6 D3 87 DD 87 96 8B 84 : 09
41C8 C4 24 02 8A 01 97 8B 39 : D0
41D0 DC 99 D3 87 DD 87 20 ED : 40
41D8 96 81 81 04 24 05 D6 8A : 25
41E0 E7 B6 39 26 11 9E 9B E6 : 2C
41E8 80 A6 80 9F 9B 1F 01 DC : DC
41F0 87 E7 84 A7 01 39 9E 9B : 0C
41F8 E6 80 A6 80 9F 9B 1F 01 : E6
--------------------------------
SUM: 22 3F 74 B6 F9 E4 88 03 : F3

4200 96 8A A7 84 39 96 81 84 : 1F
4208 06 81 04 24 05 E6 B6 D7 : 27
4210 8A 39 26 11 9E 9B E6 80 : 99
4218 A6 80 9F 9B 1F 01 E6 84 : EA
4220 A6 01 DD 87 39 9E 9B E6 : 63
4228 80 A6 80 9F 9B 1F 01 A6 : A6
4230 84 97 8A 39 96 81 81 06 : 7C
4238 27 0A 33 A6 EC C4 C3 00 : 7D
4240 01 ED C4 39 DC 99 C3 00 : 23
4248 01 DD 99 39 96 81 84 06 : 51
4250 81 06 27 0A 33 A6 EC C4 : 41
4258 83 00 01 ED C4 39 DC 99 : E3
4260 83 00 01 DD 99 39 96 81 : 4A
4268 81 06 27 22 E6 A6 5C E7 : 9F
4270 A6 96 8B 84 01 C1 80 26 : B3
4278 02 8A 04 5D 26 03 8A 40 : E0
--------------------------------
SUM: 4F 02 C6 A2 60 B6 EE 22 : DF

4280 5D 2A 02 8A 80 C4 0F 26 : 8C
4288 02 8A 10 97 8B 39 9D 9D : 31
4290 E6 C4 5C E7 C4 20 DA 96 : 41
4298 81 81 06 27 26 E6 A6 5A : 3B
42A0 E7 A6 96 8B 84 01 8A 02 : BF
42A8 C1 7F 26 02 8A 04 5D 26 : 79
42B0 03 8A 40 5D 2A 02 8A 80 : 60
42B8 C4 0F C1 0F 26 02 8A 10 : 65
42C0 97 8B 39 9D 9D E6 C4 5A : 99
42C8 E7 C4 20 D6 D6 81 C1 06 : BF
42D0 27 09 9E 9B A6 80 9F 9B : C9
42D8 A7 A5 39 9D 9D 9E 9B A6 : 9E
42E0 80 9F 9B A7 C4 39 96 81 : 75
42E8 48 8E 12 F0 AE 86 6E 84 : FE
42F0 13 00 13 22 13 36 13 4B : EF
42F8 13 74 14 25 14 1C 14 2E : 32
--------------------------------
SUM: 6F 55 35 B1 A2 A2 11 8A : 89

4300 96 8A 8D 09 97 8A 39 96 : A6
4308 89 8D 02 20 4F 48 25 07 : FB
4310 D6 8B C4 C4 D7 8B 39 8A : 0E
4318 01 D6 8B C4 C4 CA 01 D7 : 8C
4320 8B 39 96 8A 8D 09 97 8A : 9B
4328 39 96 89 8D 02 20 2D 44 : 78
4330 24 DE 8A 80 20 E3 96 8A : 2F
4338 8D 09 97 8A 39 96 89 8D : 9C
4340 02 20 19 D6 8B 54 49 25 : 5E
4348 D0 20 C5 96 8A 8D 03 97 : FC
4350 8A 39 D6 8B 54 46 20 EF : CD
4358 96 89 8D F6 D6 82 C1 06 : C1
4360 26 09 A7 C4 D6 8B C4 01 : C0
4368 4D 20 69 A7 A5 D6 8B C4 : 47
4370 01 4D 20 60 96 8A D6 8B : 4F
4378 C5 02 26 2E D7 89 5F 84 : 5E
--------------------------------
SUM: 96 A8 B5 B8 90 E6 2C 68 : B5

4380 0F 81 09 23 02 CA 10 D7 : 6F
4388 8B D6 89 C5 10 26 03 1C : 04
4390 DF 8E 1A 20 54 96 8A 19 : 34
4398 97 8A D6 8B 24 02 CA 01 : 73
43A0 84 0F 81 09 23 2C C4 EF : 1F
43A8 20 28 0F 89 C5 10 27 04 : E0
43B0 86 06 97 89 C5 01 27 06 : 9F
43B8 86 60 9A 89 97 89 C4 03 : F0
43C0 00 89 96 8A 9B 89 34 01 : 02
43C8 97 8A 35 02 85 20 27 02 : 26
43D0 CA 10 96 8A 26 03 CA 40 : 2D
43D8 4D 2A 02 CA 80 D7 89 5F : 82
43E0 44 24 02 5C 4D 26 F9 96 : C8
43E8 89 C5 01 26 02 8A 04 97 : 9C
43F0 8B 39 96 89 48 20 0F 96 : F0
43F8 89 47 20 0A 96 89 48 8A : EB
--------------------------------
SUM: 4F C2 5F 2C C1 2A 3F F8 : BE

4400 01 20 03 96 89 44 1F A9 : 4F
4408 C4 01 D7 89 D6 82 C1 06 : 44
4410 26 03 A7 C4 8E A7 A5 D6 : 44
4418 89 4D 20 B8 96 8B 84 C4 : 17
4420 8A 01 97 8B 39 03 8A 96 : 09
4428 8B 8A 12 97 8B 39 96 8B : A3
4430 85 01 26 07 84 C4 8A 01 : 86
4438 97 8B 39 84 C4 97 8B 39 : FE
4440 96 81 D6 82 81 06 27 10 : 2D
4448 C1 06 27 05 E6 A5 E7 A6 : 0B
4450 39 9D AF E6 C4 E7 A6 39 : F5
4458 C1 06 27 08 1F 98 9D AF : F9
4460 A6 A6 A7 C4 39 96 82 81 : 89
4468 06 26 05 9D 9D E6 C4 8E : A3
4470 E6 A6 D7 89 96 81 48 8E : D9
4478 14 7E AE 86 6E 84 14 96 : 62
--------------------------------
SUM: 9C A2 AD 2D B3 3A 31 75 : AB

4480 14 BE 14 C3 14 E5 15 0E : C5
4488 15 23 15 2C 15 35 9E 9B : FC
4490 E6 80 9F 9B 20 DC 1C FE : B6
4498 D6 8A D9 89 1F A8 D7 8A : EA
44A0 5F 85 20 27 02 CA 10 44 : 4B
44A8 24 02 CA 01 44 24 02 CA : 25
44B0 04 44 24 02 CA 40 44 24 : E0
44B8 02 CA 80 D7 8B 39 D6 8B : 48
44C0 54 20 D5 C6 02 D7 8B D6 : 49
44C8 8A D7 80 D0 89 1F A8 D7 : D8
44D0 8A D6 89 C4 0F D7 89 D6 : F2
44D8 80 C4 0F D1 89 D6 8B 24 : 32
44E0 C6 CA 10 20 C2 D6 8B C5 : A8
44E8 01 27 D8 D6 8A C4 0F 27 : 5A
44F0 03 C6 02 8E C6 12 D7 8B : 93
44F8 D6 8A C0 01 1F A8 D7 8A : 49
--------------------------------
SUM: F6 52 C6 C4 57 FC 61 96 : 1C

4500 D6 8B 44 24 02 CA 01 44 : DA
4508 24 BB CA 04 20 B7 96 8A : A4
4510 94 89 97 8A C6 10 4D 26 : 87
4518 03 CA 40 4D 2A 02 CA 80 : D0
4520 7E 13 DD 96 8A 98 89 97 : 46
4528 8A 5F 20 EA 96 8A 9A 89 : 36
4530 97 8A 5F 20 E1 C6 02 D7 : 20
4538 8B D6 8A D7 80 D0 89 1F : BA
4540 A8 20 8E 96 80 84 0F 48 : 47
4548 8E 15 4F AE 86 6E 84 15 : 2D
4550 6F 15 BE 16 15 16 70 16 : 09
4558 A6 17 81 14 8E 17 19 15 : 25
4560 6F 15 DC 16 15 16 70 16 : 27
4568 A6 17 9D 14 8E 17 19 96 : C2
4570 8B D6 81 58 8E 15 7B AE : 06
4578 85 6E 84 15 96 15 9B 15 : E7
--------------------------------
SUM: 2B 3C 65 7B 03 C1 17 81 : A3

4580 A0 15 A5 15 AA 15 AF 15 : F2
4588 B4 15 B9 9E 99 EC 81 9F : C5
4590 99 D7 9B 97 9C 39 85 40 : 3C
4598 27 F1 39 85 40 26 EC 39 : 61
45A0 85 01 27 E7 39 85 01 26 : 79
45A8 E2 39 85 04 27 DD 39 85 : 66
45B0 04 26 D8 39 85 80 27 D3 : 3A
45B8 39 85 80 26 CE 39 96 81 : 82
45C0 81 06 27 0D 33 A6 9E 99 : CB
45C8 EC 81 9F 99 E7 C4 A7 41 : 38
45D0 39 9E 99 EC 81 9F 99 D7 : EC
45D8 8A 97 8B 39 96 81 84 06 : 86
45E0 8E 15 E7 AE 86 6E 84 15 : C5
45E8 8B 15 EF 16 0B 16 10 10 : E6
45F0 8E 10 83 8E 10 8C C6 03 : 14
45F8 D7 89 EC A4 EE 84 ED 81 : D0
--------------------------------
SUM: 66 56 65 DA 92 99 41 8C : F3

4600 EF A1 0A 89 26 F4 10 8E : DB
4608 10 83 39 DC 87 DD 9B 39 : E0
4610 DC 87 DD 99 39 96 8B D6 : 09
4618 81 58 8E 16 21 AE 85 6E : 3F
4620 84 16 3A 16 40 16 46 16 : 9C
4628 4C 16 52 16 58 16 5E 16 : AC
4630 64 9E 9B E6 84 A6 01 DD : 8B
4638 9B 39 85 40 27 F3 20 28 : FB
4640 85 40 26 ED 20 22 85 01 : A0
4648 27 E7 20 1C 85 01 26 E1 : D7
4650 20 16 85 04 27 DB 20 10 : F1
4658 85 04 26 D5 20 0A 85 80 : B3
4660 27 CF 20 04 85 80 26 C9 : 0E
4668 DC 9B C3 00 02 DD 9B 39 : ED
4670 96 81 48 8E 16 7A AE 86 : B1
4678 6E 84 16 31 17 B0 11 88 : 99
--------------------------------
SUM: 83 B6 8C 0B EA 69 50 BE : 31

4680 11 88 16 8A 16 9D 11 5C : 59
4688 11 5C 9E 99 E6 84 A6 01 : B5
4690 9E 87 DD 87 1F 10 9E 99 : EF
4698 E7 84 A7 01 39 9E 87 DC : 4D
46A0 85 DD 87 9F 85 39 96 81 : 5D
46A8 48 8E 16 B0 AE 86 6E 84 : C2
46B0 16 D3 16 DB 16 E3 16 EB : D4
46B8 16 F3 16 FB 17 03 17 0B : 56
46C0 9E 9B E6 80 A6 80 DD 9B : 3D
46C8 DE 99 1F 10 A7 C2 E7 C2 : B8
46D0 DF 99 39 96 8B 85 40 27 : BE
46D8 E7 20 36 96 8B 85 40 26 : 49
46E0 DF 20 2E 96 8B 85 01 27 : FB
46E8 D7 20 26 96 8B 85 01 26 : EA
46F0 CF 20 1E 96 8B 85 04 27 : DE
46F8 C7 20 16 96 8B 85 04 26 : CD
--------------------------------
SUM: 2E 8D FD E4 43 D4 5B 11 : 1F

4700 BF 20 0E 96 8B 85 80 27 : 3A
4708 B7 20 06 96 8B 85 80 26 : 29
4710 AF DC 9B C3 00 02 DD 9B : 63
4718 39 96 81 81 01 27 14 8E : 9B
4720 17 2B E6 86 4F 9E 9B DD : 13
4728 9B 20 9D 00 08 10 18 20 : A8
4730 28 30 38 35 10 9E 9B A6 : B4
4738 80 9F 9B DE 99 B7 00 82 : 6A
4740 1F 10 A7 C2 E7 C2 8E 10 : DF
4748 99 C6 02 8D 27 8E 10 89 : 3C
4750 C6 03 8D 20 D6 8A 96 8B : F7
4758 ED C3 8E 10 92 C6 03 8D : 36
4760 13 D6 93 96 94 ED C3 ED : 43
4768 C3 DF 99 35 08 1F 30 D7 : 9E
4770 80 97 81 39 D7 89 A6 82 : 59
4778 E6 82 ED C3 0A 89 26 F6 : C7
--------------------------------
SUM: 5F 36 E4 4F 0A F4 35 88 : 83

4780 39 96 81 81 06 27 0B EC : F5
4788 A6 9E 99 A7 82 E7 82 9F : 0E
4790 99 39 D6 8A 96 8B 9E 99 : 8A
4798 ED 83 9F 99 39 96 81 84 : 7C
47A0 06 8E 17 A8 AE 86 6E 84 : 79
47A8 16 C0 18 3B 18 99 18 56 : 48
47B0 9D CB 34 04 96 82 81 06 : 3F
47B8 26 05 9D 9D E6 C4 8E E6 : 83
47C0 A6 D7 89 35 02 8E 17 CC : AE
47C8 AE 86 6E 84 17 D4 17 EE : 16
47D0 18 16 18 0B 96 81 48 8E : 3E
47D8 17 DE AE 86 6E 84 13 07 : 35
47E0 13 29 13 3D 13 58 13 F2 : FC
47E8 13 F7 13 FC 14 03 96 89 : 4F
47F0 D6 81 8E 18 2B A5 85 26 : 78
47F8 09 D6 8B C4 01 CA 50 D7 : 20
--------------------------------
SUM: CC D6 8B 2E 09 C5 48 35 : A6

4800 8B 39 D6 8B C4 01 CA 10 : C4
4808 D7 8B 39 96 89 D6 81 8E : 9F
4810 18 2B AA 85 20 09 96 89 : BA
4818 D6 81 8E 18 33 A4 85 D6 : 2F
4820 82 C1 06 26 03 A7 C4 39 : 16
4828 A7 A5 39 01 02 04 08 10 : A4
4830 20 40 80 FE FD FB F7 EF : BC
4838 DF BF 7F 0C 92 9D CB C1 : E4
4840 02 27 4C 9E 95 DE 87 9F : AC
4848 87 DF 95 8D 24 9E 95 DE : BD
4850 87 9F 87 DF 95 39 03 92 : EF
4858 9D CB C1 02 27 31 9E 97 : B8
4860 DE 87 9F 87 DF 97 8D 09 : 97
4868 9E 97 DE 87 9F 87 DF 97 : 36
4870 39 96 80 81 CB 26 18 9E : 77
4878 9B EC 84 E7 84 A7 01 34 : 52
--------------------------------
SUM: 75 E5 2F 71 76 98 36 0E : 4C

4880 10 BD 17 B0 35 10 EC 84 : 49
4888 E7 84 A7 01 0F 92 39 8E : 7B
4890 11 0E AE 85 AD 84 0F 92 : 24
4898 39 9D CB C1 04 10 27 00 : 9D
48A0 E9 96 82 48 8E 18 AB AE : 48
48A8 86 6E 84 11 5C 11 5C 18 : 6A
48B0 BB 19 52 19 41 15 8B 11 : 31
48B8 5C 1A 9B D6 81 C5 01 1F : 4D
48C0 A8 C4 06 C1 06 26 03 DE : 40
48C8 99 8E EE A5 DF 80 1F 8A : C2
48D0 27 42 D6 8B 54 24 17 DC : 35
48D8 87 C3 00 01 34 01 DD 87 : E4
48E0 35 02 5F 44 24 02 CA 01 : CB
48E8 44 24 04 CA 04 21 5F D7 : 91
48F0 8B DC 87 D3 80 34 01 DD : 53
48F8 87 35 02 D6 8B 44 24 02 : 89
--------------------------------
SUM: 41 B1 E0 E8 41 9F 52 1C : 08

4900 CA 01 44 24 02 CA 04 44 : 47
4908 24 02 CA 40 44 24 02 CA : 64
4910 80 D7 8B 39 D6 8B 54 24 : F4
4918 18 DC 87 83 00 01 34 01 : 34
4920 DD 87 35 02 C6 02 44 24 : CB
4928 02 CA 01 44 24 05 CA 04 : 08
4930 8E C6 02 D7 8B DC 87 93 : AE
4938 80 34 01 DB 87 35 02 20 : 70
4940 BA D6 8A 00 8A 1F A8 C4 : 2F
4948 0F 26 03 5F 20 AF C6 20 : 4C
4950 20 AB D6 81 C5 01 34 01 : 1D
4958 C4 06 C1 06 26 04 CE 10 : 99
4960 99 8E 33 A5 35 01 26 11 : 6C
4968 9E 9B E6 80 A6 80 9F 9B : FF
4970 1F 01 EC C4 E7 84 A7 01 : E3
4978 39 9E 9B E6 80 A6 80 9F : 9D
--------------------------------
SUM: AF 76 1D CF EF 10 81 4F : E0

4980 9B 1F 01 EC 84 E7 C4 A7 : 7D
4988 41 39 D6 82 C5 02 27 01 : C1
4990 39 C5 01 27 14 D6 81 C4 : 55
4998 03 58 8E 19 A1 AE 85 6E : 44
49A0 84 19 BD 19 C7 1A 08 1A : 76
49A8 15 D6 81 C4 03 58 8E 19 : 32
49B0 B5 AE 85 6E 84 1A 22 1A : 30
49B8 30 1A 51 1A 72 9E 87 E6 : 32
49C0 80 D7 89 9F 87 20 0A 9E : CE
49C8 87 E6 84 D7 89 30 1F 9F : 3F
49D0 87 96 8A 97 80 91 89 1F : F7
49D8 A8 D6 89 C4 0F D7 89 D6 : 10
49E0 80 C4 0F D1 89 D6 8B C4 : D2
49E8 01 CA 02 24 02 CA 10 85 : 52
49F0 08 27 02 CA 80 85 04 27 : 2B
49F8 02 CA 40 9E 83 30 1F 9F : 1B
--------------------------------
SUM: 57 D4 ED 41 EB A4 29 4E : 5F

4A00 83 27 02 CA 04 D7 8B 39 : 15
4A08 8D B3 D6 8B C5 40 26 04 : D0
4A10 C5 04 26 F4 39 8D B0 D6 : 2F
4A18 8B C5 40 26 F7 C5 04 26 : 9C
4A20 F4 39 9E 85 DE 87 A6 C0 : 1B
4A28 A7 80 9F 85 DF 87 20 10 : E1
4A30 9E 85 DE 87 A6 C4 A7 84 : 1D
4A38 30 1F 33 5F 9F 85 DF 87 : 6B
4A40 D6 8B C4 C1 9E 83 30 1F : 56
4A48 9F 83 27 02 CA 04 D7 8B : 7B
4A50 39 9E 83 DE 85 10 9E 87 : F2
4A58 A6 A0 A7 C0 30 1F 26 F8 : 1A
4A60 9F 83 DF 85 10 9F 87 10 : CC
4A68 8E 10 83 96 8B 84 C1 97 : 1E
```

▶ローディストとはロードランナーの変種だと思っていた無知な人間がここにいます。
川崎　哲哉 (17) 東京都

```
4A70 8B 39 9E 83 DE 85 10 9E : F6
4A78 87 31 21 33 41 A6 A2 A7 : 3C
---------------------------------
SUM: 5C 49 C2 91 D2 C4 76 29 : 2D

4A80 C2 30 1F 26 F8 31 3F 33 : D2
4A88 5F 9F 83 DF 85 10 9F 87 : 1B
4A90 10 8E 10 83 96 8B 84 C1 : 97
4A98 97 8B 39 D6 81 C5 04 27 : A2
4AA0 F9 C5 02 26 F5 C5 01 27 : C8
4AA8 2F 9E 87 96 8A 1F 89 84 : A0
4AB0 F0 97 89 68 84 59 68 84 : 41
4AB8 59 68 84 59 68 84 59 1F : 02
4AC0 98 84 0F 9A 89 97 8A C4 : 33
4AC8 F0 54 54 54 54 EA 84 E7 : 95
4AD0 84 D6 8B C4 01 7E 15 16 : 53
4AD8 9E 87 96 8A 1F 89 C4 F0 : A1
4AE0 D7 89 44 66 84 46 66 84 : BE
4AE8 46 66 84 46 66 84 46 44 : EA
4AF0 44 44 44 9A 89 97 8A D6 : E6
4AF8 8B C4 01 7E 15 16       : F9
---------------------------------
SUM: CF 76 12 DB 84 51 CE 3F : 14
```

●カード→エミュレータ変更

```
4030 C5 AD CB BD 10 E9 20 DD : F0
4038 BD 10 F1 20 D8 34 0A 4F : 43
4040 1F 8B 17 F8 4E 35 0A 3B : 81
---------------------------------
SUM: A1 48 D3 D5 36 52 34 67 : B4
```

リスト3　DOSモジュール，ディスクI/O

```
5100 ED 7B 6C 1F CD D6 1F 3E : F3
5108 23 CD F4 1F ED 5B 76 1F : E0
5110 CD D3 1F CD 1B 21 DC 33 : D7
5118 20 18 E5 1A FE 23 28 02 : 82
5120 B7 C9 13 1A 13 B7 C8 FE : 3D
5128 21 CA 36 20 FE 4A CA 72 : C5
5130 21 FE 4C CA E1 21 FE 4B : 80
5138 CA 38 22 FE 4E CA 71 22 : CD
5140 FE 4D CA 82 21 FE 57 CA : D7
5148 82 22 FE 53 28 08 FE 44 : 67
5150 28 12 3E 0D 37 C9 1A CD : 6C
5158 AA 22 13 FE 54 CA 43 22 : 60
5160 1B C3 92 21 1A CD AA 22 : 44
5168 13 FE 56 CA 5C 22 1B C3 : 8D
5170 85 21 CD 94 22 CD B2 1F : C7
5178 3E 0D D8 EB 21 00 21 E3 : 33
---------------------------------
SUM: 03 8E C1 71 A0 B6 E4 53 : 50

5180 EB E9 C3 8E 1F CD 94 22 : C7
5188 CD 9A 22 32 5D 1F CD 06 : 0A
5190 20 C9 CD 94 22 3E 01 CD : 78
5198 A3 1F 1A FE 3A 20 3E 13 : 85
51A0 CD B2 1F 38 38 22 70 1F : BF
51A8 22 6E 1F 13 CD B2 1F 38 : 98
51B0 2C D5 ED 5B 70 1F B7 ED : 7C
51B8 52 D1 38 21 23 22 72 1F : 52
51C0 13 CD B2 1F 38 03 22 6E : 7C
51C8 1F CD AF 1F D8 CD AC 1F : 2A
51D0 D8 CD EB 1F 11 F3 2A CD : AA
51D8 E8 1F C3 EB 1F 3E 0D 37 : 56
51E0 C9 3E 01 CD A3 1F 1A B7 : 68
51E8 32 22 22 28 09 13 CD B2 : 39
51F0 1F 38 EA 22 20 22 CD 09 : 7B
51F8 20 D8 C4 23 22 20 F7 CD : E5
---------------------------------
SUM: 14 27 0F 9B 9E D4 08 3B : 9A

5200 E2 1F 4C 6F 61 64 69 6E : 58
5208 67 20 00 CD 9D 1F CD EB : C8
5210 1F 3A 22 22 B7 28 06 2A : AC
5218 20 22 22 70 1F C3 A6 1F : 7B
5220 00 00 00 F5 CD E2 1F 46 : 09
5228 6F 75 6E 64 20 20 20 00 : 16
5230 CD 9D 1F CD EB 1F F1 C9 : 1A
5238 CD 94 22 CD A3 1F D8 CD : B7
5240 15 20 C9 CD 94 22 CD A3 : F1
5248 1F 13 CD 94 22 1A FE 50 : 1D
5250 CA 0C 20 FE 52 CA 0F 20 : 3F
5258 3E 0D 37 C9 CD 94 22 1A : E8
5260 CD AA 22 CD 15 29 30 03 : D7
5268 3E 03 C9 32 5D 1F C3 27 : A2
5270 20 CD 94 22 CD A3 1F 1A : 4C
5278 13 FE 3A CA 12 20 3E 0D : 92
---------------------------------
SUM: 0B 05 E5 D4 75 53 36 FC : C3

5280 37 C9 3A 5C 1F FE 50 20 : 23
5288 05 3E 28 C3 30 20 3E 50 : 0C
5290 C3 30 20 13 1A FE 20 28 : 86
5298 FA C9 CD 94 22 13 1A 1B : 8E
52A0 FE 3A 28 03 C3 AD 25 1A : 12
52A8 13 13 FE 61 D8 FE 7B D0 : A6
52B0 E6 DF C9 CD 75 25 3A 5D : 8C
52B8 1F CD 51 28 D8 CA 06 29 : 36
52C0 CD 91 25 30 01 C9 CD FF : 49
52C8 26 D8 CD 6B 27 20 16 7E : 11
52D0 CD 7C 25 D8 CD 84 25 D8 : 94
52D8 E5 01 1E 00 09 7E E1 CD : 39
52E0 4E 27 D8 18 06 CD A2 27 : 01
52E8 3E 09 D8 ED 53 DF 27 22 : 87
52F0 E1 27 CD 3F 29 CD 70 25 : 9F
52F8 AF C9 CD 75 25 3A 5D 1F : 95
---------------------------------
SUM: D0 FF 0E 4B 18 67 27 D2 : A0

5300 CD 51 28 D8 CA 03 29 CD : E1
5308 91 25 30 01 C9 CD 6B 27 : 0F
5310 D8 3E 08 37 C0 E5 ED 5B : 42
5318 74 1F 01 20 00 ED B0 E1 : 32
5320 7E CD 84 25 D8 CD 2A 29 : EC
5328 CD 70 25 AF C9 3A 5D 1F : 90
5330 CD 51 28 D8 CA 09 29 3A : 54
5338 1E 29 B7 20 04 37 3E 0C : A3
5340 C9 CD 75 25 3A 5D 1F CD : B3
5348 91 25 D8 CD 5C 26 C9 3A : E0
5350 5D 1F CD 51 28 D8 CA 0C : 70
5358 29 AF 32 67 1F 32 18 24 : FE
5360 3A 1E 29 B7 20 04 37 3E : D1
5368 0C C9 CD 75 25 3A 5D 1F : F2
5370 CD 91 25 D8 CD FF 26 D8 : 25
5378 CD E3 25 C9 CD 75 25 3A : 3F
---------------------------------
SUM: A0 A5 75 73 7E 28 C8 64 : FF

5380 5D 1F CD 51 28 D8 20 09 : C3
5388 CD B4 25 32 5D 1F C3 00 : 17
5390 29 CD D0 1F FE 1B CA 0D : D5
5398 24 FE 0D 20 06 3A 18 24 : CB
53A0 B7 20 5F 3A 67 1F 4F 06 : 4B
53A8 03 CB 3F 10 FC 2A 60 1F : C2
53B0 16 00 5F 19 EB 2A 64 1F : 26
53B8 3E 01 CD 44 25 38 3D 79 : 63
53C0 E6 07 06 05 87 10 FD 2A : B6
53C8 64 1F 85 6F 30 01 24 7E : 4A
53D0 B7 28 13 FE FF 28 36 ED : 3A
53D8 5B 74 1F 01 20 00 ED B0 : AC
53E0 CD EC 23 C3 25 23 CD EC : A0
53E8 23 30 A6 C9 21 67 1F 34 : 9D
53F0 7E 21 66 1F BE 28 16 32 : 52
53F8 18 24 B7 C9 F5 CD 0D 24 : AF
---------------------------------
SUM: 67 AD 3C 50 CB AF 68 B2 : 34

5400 F1 C9 21 67 1F 7E B7 28 : BE
5408 01 35 AF 18 04 AF 32 67 : 49
5410 1F 32 18 24 3E 08 37 C9 : D3
5418 00 3A 5D 1F CD 51 28 D8 : D4
5420 CA 0F 29 CD 91 25 D8 CD : 2A
5428 FF 26 D8 3E 24 CD F4 1F : 3F
5430 CD 21 27 CD C1 1F 11 99 : 6C
5438 28 CD E5 1F 06 10 ED 5B : 57
5440 60 1F 2A 64 1F 3E 01 CD : 38
5448 44 25 D8 CD 54 24 C8 13 : 61
5450 10 F0 AF C9 C5 D5 06 08 : 20
5458 7E B7 28 0F FE FF 28 12 : A3
5460 CD E3 27 CD EE 1F CD C7 : 45
5468 1F 72 24 11 20 00 19 10 : 0F
5470 E7 3E AF D1 C1 B7 C9 3A : 20
5478 5D 1F CD 9C 25 D8 CD 91 : 40
---------------------------------
SUM: 31 2A F2 0D D4 8B 85 AC : EA

5480 25 D8 CD FF 26 D8 CD 6B : FF
5488 27 D8 3E 08 37 C0 7E CD : 87
5490 7C 25 D8 36 00 E5 01 1E : B3
5498 00 09 7E E1 CD 4E 27 D8 : 82
54A0 2A 64 1F 3E 01 CD 5A 25 : 38
54A8 D4 10 27 C9 3A 5D 1F CD : 57
54B0 9C 25 D8 CD 91 25 D8 D5 : C9
54B8 CD 6B 27 ED 53 DF 27 22 : C7
54C0 E1 27 D1 D8 3E 08 37 C0 : EE
54C8 7E CD 7C 25 D8 3A 5D 1F : 7A
54D0 F5 CD A3 1F F1 32 5D 1F : 23
54D8 CD 6B 27 D8 3E 0A 37 C8 : 7E
54E0 ED 5B DF 27 2A 64 1F 3E : 39
54E8 01 CD 44 25 D8 2A 74 1F : CC
54F0 23 ED 5B E1 27 13 01 11 : 98
54F8 00 ED B0 ED 5B DF 27 2A : 15
---------------------------------
SUM: 61 10 EB ED 12 F7 CE 75 : 95

5500 64 1F 3E 01 CD 5A 25 C9 : D7
5508 3A 5D 1F CD 9C 25 D8 CD : E9
5510 91 25 D8 CD 6B 27 D8 3E : 03
5518 08 37 C0 CB F6 2A 64 1F : 6D
5520 3E 01 CD 5A 25 C9 3A 5D : EB
5528 1F CD 9C 25 D8 CD 91 25 : 08
5530 D8 CD 6B 27 D8 3E 08 37 : 8C
5538 C0 CB B6 2A 64 1F 3E 01 : 2D
5540 CD 5A 25 C9 08 3A 5D 1F : D3
5548 CD 9C 25 D8 CD 91 25 D8 : C1
5550 D6 41 32 06 2B 08 CD 00 : 4F
5558 2B C9 08 3A 5D 1F CD 9C : 1B
5560 25 D8 CD 91 25 D8 D6 41 : 6F
5568 32 06 2B 08 CD 03 2B C9 : 2F
5570 F5 3E 01 18 02 F5 AF 32 : 24
5578 1E 29 F1 C9 B7 CB 77 C8 : C2
---------------------------------
SUM: 31 83 ED 91 0B 50 8D 44 : 5E

5580 3E 04 37 C9 E5 E6 87 21 : B5
5588 1F 29 BE E1 C8 3E 06 37 : 2A
5590 C9 FE 41 38 04 FE 45 3F : C6
5598 D0 3E 0B C9 CD 51 28 D8 : 00
55A0 CD 63 28 20 04 3E 03 37 : F4
55A8 C9 CD 91 25 C9 3A 20 29 : 98
55B0 CD 63 28 C0 3A 7D 1F B7 : A5
55B8 20 02 3E 54 FE 01 20 02 : D5
55C0 3E 53 FE 03 20 02 3E 51 : 43
55C8 C9 F5 32 20 29 FE 54 20 : AB
55D0 01 AF FE 53 20 02 3E 01 : 62
55D8 FE 51 20 02 3E 03 32 7D : 61
55E0 1F F1 C9 2A 74 1F 01 1E : B5
55E8 00 09 7E 32 DE 27 ED 4B : F6
55F0 72 1F 2A 70 1F E5 3A DE : 47
55F8 27 2A 62 1F 5F 16 00 19 : 60
---------------------------------
SUM: 37 89 81 67 FA AF 86 D7 : AE

5600 7E 32 DE 27 EB 29 29 29 : 1B
5608 29 EB E1 B7 28 19 FE 80 : 6B
5610 30 19 3E 10 CD 44 25 D8 : A5
5618 11 00 10 19 E5 69 60 B7 : 9F
5620 ED 52 4D 44 E1 30 CE 3E : ED
5628 07 37 C9 D6 7F FE 11 30 : 9B
5630 F6 3D 0B B8 20 F1 06 00 : 0D
5638 03 B7 28 07 F5 CD 44 25 : 14
5640 38 14 F1 D5 1E 00 57 19 : A0
5648 E3 5F 16 00 19 EB 2A 64 : EA
5650 1F 3E 01 CD 44 25 D1 D8 : 3D
5658 ED B0 AF C9 ED 5B DF 27 : 63
5660 2A E1 27 ED 4B 72 1F C5 : C0
5668 0B CB 38 CB 38 CB 38 CB : DF
5670 38 04 CD 21 27 B8 C1 3E : 08
5678 09 D8 2A 74 1F E5 D5 C5 : 1D
---------------------------------
SUM: 72 9C 63 98 6B 20 F3 DA : 61

5680 11 18 00 19 5D 54 13 36 : 3C
5688 00 01 07 00 ED B0 C1 D1 : 37
5690 E1 3E 1E 85 6F 30 01 24 : 86
5698 CD 36 27 77 2A 70 1F E5 : 3F
56A0 2A 62 1F 5F 16 00 19 EB : 24
56A8 29 29 29 29 EB 0B 78 03 : 15
56B0 FE 10 38 21 36 80 CD 36 : 20
56B8 27 77 E1 F5 3E 10 CD 5A : E9
56C0 25 38 10 11 00 10 19 E5 : 8C
56C8 69 60 B7 ED 52 4D 44 E1 : 31
56D0 F1 18 CC E1 C9 3C F5 C6 : 76
56D8 7F 77 F1 E1 CD 5A 25 D8 : EC
56E0 CD 10 27 D8 2A 74 1F ED : 86
56E8 5B E1 27 01 20 00 ED B0 : 21
56F0 2A 64 1F ED 5B DF 27 3E : 39
56F8 01 CD 5A 25 D8 AF C9 D5 : 72
---------------------------------
SUM: 88 E8 F8 5E BD 34 92 A2 : EB

5700 E5 ED 5B 5E 1F 2A 62 1F : 55
5708 3E 01 CD 44 25 E1 D1 C9 : F0
5710 D5 E5 ED 5B 5E 1F 2A 62 : 0B
5718 1F 3E 01 CD 5A 25 E1 D1 : 5C
5720 C9 C5 E5 06 80 0E 00 2A : 31
5728 62 1F 7E B7 20 01 0C 23 : 06
5730 10 F8 79 E1 C1 C9 C5 E5 : 96
5738 06 80 2A 62 1F 7E B7 28 : 8E
5740 06 23 10 F9 37 18 04 3E : C3
5748 80 90 B7 E1 C1 C9 D5 E5 : EC
5750 ED 5B 62 1F 6F 26 00 19 : 77
5758 7E 36 00 FE 80 38 F5 E1 : 40
5760 D1 FE 90 30 02 AF C9 3E : 47
5768 07 37 C9 C5 0E 10 ED 5B : 32
5770 60 1F 2A 64 1F 3E 01 CD : 38
5778 44 25 38 24 06 08 7E FE : 4F
---------------------------------
SUM: C5 2A 00 3E 98 E9 C9 F6 : 6D

5780 FF 28 1A B7 28 0B D5 ED : ED
5788 5B 74 1F CD CD 27 D1 28 : A8
5790 0D D5 11 20 00 19 D1 10 : 0D
5798 E5 13 0D 20 D5 3E AF B7 : 9E
57A0 C1 C9 C5 0E 10 ED 5B 60 : 15
57A8 1F 2A 64 1F 3E 01 CD 44 : 1C
57B0 25 38 16 06 08 7E B7 28 : DE
57B8 11 FE FF 28 0D D5 11 20 : 49
57C0 00 19 D1 10 F0 13 0D 20 : 2A
57C8 E0 3E AF C1 C9 C5 D5 E5 : D6
57D0 06 10 13 23 1A BE 20 02 : 46
57D8 10 F8 E1 D1 C1 C9 00 00 : 44
57E0 00 00 00 C5 D5 E5 ED 5B : C7
57E8 74 1F 01 20 00 ED B0 CD : 1E
57F0 27 28 3A 5D 1F CD F4 1F : E5
57F8 3E 3A CD F4 1F CD 9D 1F : E1
---------------------------------
```

▶MZ-2500用拡張RAMボードの価格改定は，私たちにとって実に喜ばしいことである。
渡辺　亮治（17）群馬県

```
SUM: 31 8D 11 1A D4 95 46 35 : CD

5800 CD 2A 29 ED 4B 72 1F 2A : 13
5808 70 1F ED 5B 6E 1F CD 1E : 4F
5810 28 09 2B CD 1E 28 EB CD : 27
5818 1E 28 E1 D1 C1 C9 3E 3A : FA
5820 CD F4 1F CD BE 1F C9 F5 : 48
5828 11 A9 28 CB 7F 28 03 3E : 95
5830 08 11 E6 07 6F 26 00 29 : C4
5838 29 11 A9 28 19 EB CD E5 : C1
5840 1F F1 CB 77 3E 2A 20 02 : DC
5848 3E 20 CD F4 1F CD F1 1F : 1B
5850 C9 CD 63 28 C8 FE 41 38 : 60
5858 07 FE 4D 3F 38 02 B7 C9 : 4B
5860 3E 03 C9 FE 54 C8 FE 53 : 75
5868 C8 FE 51 C9 3D FE 0E 38 : 61
5870 13 3C 11 E3 2A F5 CD E8 : 17
5878 1F 3E 24 CD F4 1F F1 CD : 1F
-----------------------------------
SUM: F7 90 8F F6 69 AB 81 F2 : 93

5880 C1 1F 18 0E 21 00 2A 87 : D8
5888 5F 16 00 19 5E 23 56 CD : 32
5890 E8 1F CD C4 1F CD EB 1F : 8E
5898 C9 20 43 6C 75 73 74 65 : 59
58A0 72 73 20 46 72 65 65 0D : 94
58A8 00 4E 75 6C 00 42 69 6E : 48
58B0 00 42 61 73 00 3F 3F 3F : D3
58B8 00 41 73 63 00 3F 3F 3F : D4
58C0 00 3F 3F 3F 00 3F 3F 3F : 7A
58C8 00 44 69 72 00 00 00 00 : 1F
58D0 00 00 00 00 00 00 00 00 : 00
58D8 00 00 00 00 00 00 00 00 : 00
58E0 00 00 00 00 00 00 00 00 : 00
58E8 00 00 00 00 00 00 00 00 : 00
58F0 00 00 00 00 00 00 00 00 : 00
58F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 43 3B 39 90 85 C7 6A 10 : 0D

5900 C3 E4 1D C3 E7 1D C3 EA : 38
5908 1D C3 ED 1D C3 F0 1D C3 : 7D
5910 F3 1D C3 E3 27 C3 51 28 : 19
5918 C3 63 28 00 00 00 00 00 : 4E
5920 41 00 00 00 00 00 00 00 : 41
5928 00 00 E5 2A 12 07 22 72 : BC
5930 1F 2A 14 07 22 70 1F 2A : 3F
5938 16 07 22 6E 1F E1 C9 E5 : 5B
5940 2A 72 1F 22 12 07 2A 70 : 90
5948 1F 22 14 07 2A 6E 1F 22 : 35
5950 16 07 E1 C9 00 00 00 00 : C7
5958 00 00 00 00 00 00 00 00 : 00
5960 00 00 00 00 00 00 00 00 : 00
5968 00 00 00 00 00 00 00 00 : 00
5970 00 00 00 00 00 00 00 00 : 00
5978 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 6B F3 24 54 60 9D 84 E8 : 3F

5980 00 00 00 00 00 00 00 00 : 00
5988 00 00 00 00 00 00 00 00 : 00
5990 00 00 00 00 00 00 00 00 : 00
5998 00 00 00 00 00 00 00 00 : 00
59A0 00 00 00 00 00 00 00 00 : 00
59A8 00 00 00 00 00 00 00 00 : 00
59B0 00 00 00 00 00 00 00 00 : 00
59B8 00 00 00 00 00 00 00 00 : 00
59C0 00 00 00 00 00 00 00 00 : 00
59C8 00 00 00 00 00 00 00 00 : 00
59D0 00 00 00 00 00 00 00 00 : 00
59D8 00 00 00 00 00 00 00 00 : 00
59E0 00 00 00 00 00 00 00 00 : 00
59E8 00 00 00 00 00 00 00 00 : 00
59F0 00 00 00 00 00 00 00 00 : 00
59F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 00 00 00 00 00 00 00 00 : 00

5A00 1C 2A 2D 2A 3C 2A 50 2A : 7D
5A08 60 2A 6B 2A 79 2A 8E 2A : 7A
5A10 9D 2A A9 2A BD 2A CE 2A : 79
5A18 DC 2A EA 2A 44 65 76 69 : A2
5A20 63 65 20 49 2F 4F 20 45 : 14
5A28 72 72 6F 72 0D 44 65 76 : F1
5A30 69 63 65 20 4F 66 66 6C : D8
5A38 69 6E 65 0D 42 61 64 20 : 70
5A40 46 69 6C 65 20 44 65 73 : BC
5A48 63 72 69 70 74 65 72 0D : 06
5A50 57 72 69 74 65 20 50 72 : ED
5A58 6F 74 65 63 74 65 64 0D : F5
5A60 42 61 64 20 52 65 63 6F : B0
5A68 72 64 0D 42 61 64 20 46 : 50
5A70 69 6C 65 20 4D 6F 64 65 : DF
5A78 0D 42 61 64 20 41 6C 6C : 4D
-----------------------------------
SUM: 35 84 5E 22 10 E4 4F B3 : 2F

5A80 6F 63 61 74 69 6F 6E 20 : 0D
5A88 54 61 62 6C 65 0D 46 69 : A4
5A90 6C 65 20 6E 6F 74 20 46 : A8
5A98 6F 75 6E 64 0D 44 65 76 : E2
5AA0 69 63 65 20 46 75 6C 6C : E4
5AA8 0D 46 69 6C 65 20 41 6C : 5A
5AB0 72 65 61 64 79 20 45 78 : F2
5AB8 69 73 74 73 0D 52 65 73 : FA
5AC0 65 72 76 65 64 20 46 65 : E1
5AC8 61 74 75 72 65 0D 46 69 : DD
5AD0 6C 65 20 6E 6F 74 20 4F : B1
5AD8 70 65 6E 0D 53 79 6E 74 : FE
5AE0 61 78 20 45 72 72 6F 72 : 03
5AE8 20 0D 42 61 64 20 44 61 : F9
5AF0 74 61 0D 43 6F 6D 70 6C : DD
5AF8 65 74 65 20 21 0D 00 00 : 8C
-----------------------------------
SUM: EB 29 41 70 6C 61 CD D8 : 37

5B00 C3 07 2B C3 10 2B 00 C5 : B8
5B03 ED 4B 06 2B CF 1A C1 C9 : DC
5B10 C5 ED 4B 06 2B CF 1B C1 : D9
5B18 C9                      : C9
-----------------------------------
SUM: 3E 3F 7C F4 0A 14 DC 4F : 36
```

リスト4　メモリエディタ

```
6000 11 30 32 CD E5 1F ED 5B : 8C
6008 76 1F CD D3 1F 1A FE 1B : 87
6010 C8 CD B2 1F 38 EA 22 7B : 25
6018 32 11 43 32 CD E5 1F CD : 56
6020 27 30 CD D4 30 18 D9 21 : 3A
6028 00 04 CD 1E 20 2A 7B 32 : E6
6030 0E 10 CD 3D 30 CD EE 1F : 32
6038 0D 20 F7 18 24 E5 CD BE : D0
6040 1F E1 06 08 1E 00 CD F1 : EA
6048 1F 7E 23 57 CD C1 1F 7A : 3E
6050 83 5F 10 F2 CD F1 1F 3E : FF
6058 3A CD F4 1F 7B CD C1 1F : 42
6060 C9 21 00 14 CD 1E 20 06 : 0F
6068 20 3E 2D CD F4 1F 10 FB : 76
6070 CD EE 1F 11 75 32 CD E5 : 44
6078 1F 2A 7B 32 0E 08 06 10 : 22
-----------------------------------
SUM: 93 93 46 CC 24 F2 0A AC E483

6080 11 08 00 AF 86 19 10 FC : 73
6088 CD C1 1F CD F1 1F 11 7F : 1A
6090 00 B7 ED 52 0D 20 E7 3E : 48
6098 3A C9 CD F4 1F 2A 7B 32 : BA
60A0 06 80 56 5A 23 05 28 27 : AD
60A8 5E 23 05 28 22 D5 1E 80 : 43
60B0 D9 E1 D9 7E A3 28 01 37 : 14
60B8 D9 ED 6A 30 08 3E 10 AC : 62
60C0 67 3E 21 AD 6F D9 CB 0B : 91
60C8 30 E9 23 10 E6 D9 EB EB : E1
60D0 CD BE 1F C9 21 05 04 22 : BF
60D8 7D 32 2A 7D 32 CD 1E 20 : 93
60E0 CD 21 20 01 19 00 21 CA : 13
60E8 31 ED B1 20 ED FE 30 30 : 3A
60F0 34 FE 1C 38 26 20 05 CD : 9E
60F8 71 31 18 DE FE 1D 20 05 : D8
-----------------------------------
SUM: B2 0E 09 2C 65 81 28 79 296D

6100 CD 94 31 18 D5 FE 1E 20 : BB
6108 09 06 10 CD 94 31 10 FB : BC
6110 18 C8 06 10 CD 71 31 10 : 75
6118 FB 18 BF 21 7D 32 FE 0D : AD
6120 C0 36 05 18 B5 FE 54 20 : 3A
6128 0E 2A 7B 32 11 80 00 B7 : 2D
6130 ED 52 22 7B 32 18 0E FE : 32
6138 47 20 0F 2A 7B 32 11 80 : DE
6140 00 19 22 7B 32 CD 27 30 : 0C
6148 18 8A FE 52 20 05 CD 61 : 45
6150 30 18 81 CD E3 31 2A 7D : 51
6158 32 2E 00 CD 1E 20 CD 27 : 5F
6160 32 5F 16 00 2A 7B 32 19 : 97
6168 CD 3D 30 CD 71 31 C3 DA : 46
6170 30 C5 21 7D 32 3E 1B BE : DC
6178 20 05 CD 89 31 18 08 7E : 4A
-----------------------------------
SUM: B4 9B 8C 3F 77 BF D3 F1 60DD

6180 34 CD B7 31 20 01 34 C1 : FF
6188 C9 36 05 23 7E 34 FE 13 : EA
6190 C0 36 04 C9 C5 21 7D 32 : 58
6198 3E 05 BE 20 05 CD AC 31 : D0
61A0 18 08 7E 35 CD B7 31 28 : B0
61A8 01 35 C1 C9 36 1B 23 7E : B2
61B0 35 FE 04 C0 36 13 C9 E5 : EE
61B8 01 08 00 21 C2 31 ED B1 : BB
61C0 E1 C9 06 09 0C 0F 12 15 : FB
61C8 18 1B 0D 1B 1C 1D 1E 1F : D1
61D0 30 31 32 33 34 35 36 37 : 9C
61D8 38 39 41 42 43 44 45 46 : 06
61E0 54 47 52 CD 1F 32 47 CD : 1F
61E8 27 32 4F C5 3A 7D 32 D6 : 2C
61F0 05 01 03 00 91 38 03 04 : D9
61F8 18 FA 78 C1 81 5F 16 00 : 41
-----------------------------------
SUM: 43 43 63 08 6D 24 A2 CB E6FC

6200 2A 7B 32 19 3A 7D 32 C5 : 9E
6208 CD B7 31 C1 7E 20 05 E6 : FF
6210 F0 B0 18 09 E6 0F 4F 78 : 7D
6218 87 87 87 87 B1 77 C9 D6 : E3
6220 30 FE 0A D8 D6 07 C9 3A : F0
6228 7E 32 D6 04 87 87 87 C9 : E8
6230 0C 49 4E 50 55 54 20 53 : 0F
6238 54 41 52 54 20 41 44 52 : 32
6240 53 0D 00 0C 44 55 4D 50 : A2
6248 20 26 20 45 44 49 54 0D : 99
6250 0D 41 44 52 53 20 2B 30 : B2
6258 20 2B 31 20 2B 32 20 2B : 44
6260 33 20 2B 34 20 2B 35 20 : 52
6268 2B 36 20 2B 37 20 3A 53 : 90
6270 55 4D 0D 0D 00 53 55 4D : B1
6278 3A 20 00 00 90 00 00    : EA
-----------------------------------
SUM: 09 85 6F 19 0E D4 B3 19 EFDD
```

リスト5　IPL

```
2100 16 00 92 0A 00 08 00 01 : BB
2108 01 00 00 07 2A 2A 2A 2A : B0
2110 20 53 2D 4F 53 20 53 57 : 0C
2118 4F 52 44 20 49 50 4C 20 : 0A
2120 73 74 61 72 74 65 64 20 : 17
2128 21 0D 00 07 2A 2A 2A 2A : DD
2130 20 53 79 73 74 65 6D 20 : C5
2138 6D 6F 64 75 6C 65 20 6C : 12
2140 6F 61 64 69 6E 67 20 63 : F5
2148 6F 6D 70 6C 65 74 65 64 : 5A
2150 20 21 0D 00 07 11 02 2A : 92
2158 2A 2A 2A 20 49 2F 4F 20 : 85
2160 66 61 75 6C 74 20 64 65 : 05
2168 74 65 63 74 65 64 20 21 : BA
2170 00 96 05 2B FC 86 80 97 : 5F
2178 05 96 05 2A FC 39 8D F1 : 7D
-----------------------------------
SUM: AE F3 2B 0B 38 59 4B 97 : 4D

2180 CE FC 82 CC 03 00 A7 C1 : 83
2188 5C A6 80 A7 C0 26 F9 F7 : FF
2190 FC 83 0F 05 39 1A 50 10 : 46
2198 CE FC 7F 86 FD 1F 8B 97 : 0D
21A0 0F 86 01 97 03 8D CA CC : 53
21A8 0C 3F FD FC 82 8D E3 30 : 66
21B0 8D FF 58 8D C9 30 8D FF : F6
21B8 4A EE 02 C6 28 34 54 BD : 6D
21C0 FE 08 35 54 4D 26 2C 5A : 88
21C8 27 20 33 C9 01 00 EF 02 : 35
21D0 6C 05 A6 05 81 10 23 E5 : B5
21D8 86 01 A7 05 6C 06 A6 06 : 51
21E0 81 01 23 D9 6F 06 6C 04 : 63
21E8 20 D3 30 8D FF 3D 8D 8E : 07
21F0 7E 08 00 30 8D FF 5D 8D : 2C
21F8 85 20 FE                : A3
-----------------------------------
SUM: A1 FD EE A1 A5 5B 43 7D : ED
```

リスト6　SYSGENo

```
2000 20 68 09 00 00 00 00 00 : 91
2008 00 00 21 00 00 01 00 01 : 23
2010 22 00 00 0F 00 02 24 00 : 57
2018 00 01 01 10 38 00 01 01 : 4C
2020 00 2C 00 00 00 00 00 00 : 2C
2028 34 40 30 8C D5 EC C4 ED : A2
2030 02 EC 42 ED 04 EC 44 A7 : F8
2038 06 EE 02 AD 9F FB FA 25 : 5C
2040 26 5A 27 20 33 C9 01 00 : C4
2048 EF 02 6C 05 A6 05 81 10 : 9E
2050 23 E9 86 01 A7 05 6C 06 : B1
2058 A6 06 81 01 23 DD 6F 06 : A3
2060 6C 04 20 D7 5F 35 C0 53 : 0E
2068 35 C0 33 8C 9D A6 45 27 : 63
2070 08 8D B5 25 04 33 46 20 : 0C
2078 F4 39                   : 2D
-----------------------------------
SUM: F9 84 41 F4 53 94 CF 71 : D9
```

▶ Oh！MZも50冊を超え，置くところがない。でも絶対に捨てないぞ！
渡部　圭誠（18）愛媛県

リスト7　FORMAT & SYSGEN

```
7000 AF 32 6B 73 CD E2 1F 0C : 99
7008 31 29 20 4C 6F 67 69 63 : 68
7010 61 6C 20 46 6F 72 6D 61 : E2
7018 74 0D 32 29 20 26 20 53 : 95
7020 79 73 74 65 6D 20 47 65 : FE
7028 6E 65 72 61 74 65 0D 33 : BF
7030 29 20 45 6E 64 20 6F 66 : 55
7038 20 4A 6F 62 0D 0D 49 6E : 0C
7040 70 75 74 20 4E 75 6D 62 : 0B
7048 65 72 20 00 CD 21 20 FE : 03
7050 1B C8 FE 33 20 06 3E 0C : 84
7058 CD F4 1F C9 FE 31 38 EC : FC
7060 FE 33 30 E8 32 6C 73 CD : 27
7068 F4 1F FE 31 CA 0F 72 3A : C7
7070 6B 73 B7 C2 02 71 CD E2 : 79
7078 1F 0D 0D 53 6F 75 72 63 : 45
--------------------------------
SUM: 1E 8B 1A 0E C3 C1 48 33 8429

7080 65 20 44 72 69 76 65 20 : 9F
7088 4E 61 6D 65 20 3D 20 00 : FE
7090 CD C6 72 D8 CD E9 72 D8 : DD
7098 20 DC CD E2 1F 0D 0D 4E : 32
70A0 6F 77 20 53 79 73 74 65 : 1E
70A8 6D 20 4C 6F 61 64 69 6E : E4
70B0 67 2E 2E 2E 2E 00 21 00 : 40
70B8 80 11 00 00 3E 01 CD 00 : 9D
70C0 20 DA 18 73 21 00 81 11 : 38
70C8 20 00 3E 28 CD 00 20 DA : 4D
70D0 18 73 CD 55 73 28 26 CD : 3B
70D8 C4 1F CD E2 1F 0D 0D 54 : 1F
70E0 68 69 73 20 69 73 20 6E : CE
70E8 6F 74 20 53 79 73 74 65 : 1B
70F0 6D 20 44 69 73 6B 20 21 : 59
70F8 21 00 C3 76 70 3E FF 32 : 39
--------------------------------
SUM: E4 62 14 A5 00 45 56 4B E088

7100 6B 73 CD C4 1F CD E2 1F : 5C
7108 0D 0D 53 79 73 74 65 6D : 9F
7110 20 4C 6F 61 64 20 43 6F : 72
7118 6D 70 6C 65 74 65 20 21 : C8
7120 21 0D 0D 00 CD E2 1F 0D : 16
7128 0D 44 65 73 74 69 6E 61 : D5
7130 74 69 6F 6E 20 44 72 69 : F9
7138 76 65 20 4E 61 6D 65 20 : 9C
7140 3D 20 00 CD C6 72 D8 CD : 07
7148 E9 72 D8 20 B5 ED 5B 5E : AE
7150 1F 2A 62 1F 3E 01 CD 00 : D6
7158 20 DA 18 73 11 F5 73 1A : 18
7160 B7 28 61 13 BE 23 20 02 : 56
7168 18 F5 CD E2 1F 0D 0D 4E : 43
7170 6F 77 20 53 79 73 74 65 : 1E
7178 6D 20 57 72 69 74 69 6E : 0A
--------------------------------
SUM: 2D A5 F3 6B B5 2E 8B 7B 3A8E

7180 67 2E 2E 2E 2E 0D 00 21 : 4D
7188 00 80 11 00 00 3E 01 CD : 9D
7190 03 20 DA 18 73 21 00 81 : 2A
7198 11 20 00 3E 28 CD 03 20 : 87
71A0 DA 18 73 CD E2 1F 0D 53 : 93
71A8 79 73 74 65 6D 20 47 65 : FE
71B0 6E 65 72 61 74 65 20 43 : E2
71B8 6F 6D 70 6C 65 74 65 20 : 16
71C0 21 00 18 6F CD C4 1F CD : 25
71C8 E2 1F 0D 0D 44 65 73 74 : AB
71D0 69 6E 61 74 69 6F 6E 20 : 12
71D8 69 73 20 53 79 73 74 65 : 14
71E0 6D 20 44 69 73 6B 2E 0D : 53
71E8 44 6F 20 79 6F 75 20 77 : C7
71F0 61 6E 74 20 74 6F 20 77 : DD
71F8 72 69 74 65 20 6F 6E 20 : D1
--------------------------------
SUM: 04 B1 D4 2D 5A 1A 2D 8B 782C

7200 69 74 20 3F 00 CD E9 72 : 64
7208 D8 C2 24 71 C3 6A 71 CD : 9A
7210 E2 1F 0D 0D 46 6F 72 6D : AF
7218 61 74 20 44 72 69 76 65 : EF
7220 20 4E 61 6D 65 20 3D 20 : 1E
7228 00 CD C6 72 D8 CD E9 72 : 05
7230 D8 20 DC CD E2 1F 0D 0D : BC
7238 49 6E 69 74 69 61 6C 69 : 33
7240 7A 69 6E 67 2E 2E 2E 0D : 4F
7248 00 3E 01 ED 5B 62 1F 12 : 1A
7250 13 3E 8F 12 13 0E 03 3A : 50
7258 6C 73 FE 31 28 0D 3E 03 : 84
7260 12 13 3C 12 13 3E 8F 12 : 65
7268 13 0E 06 AF 12 62 6B 13 : C8
7270 3A 66 1F 91 4F 06 00 ED : 92
7278 B0 3E 8F 23 13 77 3A 66 : CA
--------------------------------
SUM: CD 8F C9 2D 4E 44 A3 ED 2A56

7280 1F 4F 3E FF 91 4F 06 00 : 91
7288 ED B0 3E 01 ED 5B 5E 1F : A1
7290 2A 62 1F CD 03 20 DA 18 : 8D
7298 73 21 00 B0 36 FF 11 01 : 8B
72A0 B0 01 FF 0F ED B0 3E 10 : AA
72A8 ED 5B 60 1F 21 00 B0 CD : 65
72B0 03 20 38 64 CD E2 1F 0D : 9A
72B8 43 6F 6D 70 6C 65 74 65 : 39
72C0 20 21 0D 00 18 72 CD 21 : C6
72C8 20 FE 1B 20 02 37 C9 FE : 59
72D0 61 38 06 FE 65 30 EF D6 : F7
72D8 20 FE 41 38 E9 FE 45 30 : F3
72E0 E5 32 5D 1F CD F4 1F B7 : 2A
72E8 C9 CD E2 1F 0D 0D 41 72 : 64
72F0 65 20 79 6F 75 20 73 75 : EA
72F8 72 65 20 28 59 2F 4E 29 : 1E
--------------------------------
SUM: D2 46 E6 AA 0E E7 BB 73 3BBA

7300 20 3F 20 00 CD 21 20 FE : 8B
7308 1B 20 02 37 C9 CD F4 1F : 1D
7310 FE 59 C8 FE 79 37 3F C9 : D5
7318 CD EE 1F CD 33 20 CD E2 : A9
7320 1F 52 65 74 72 79 20 28 : 7D
7328 59 2F 4E 29 20 3F 20 00 : 7E
7330 CD 04 73 D8 CA 00 70 C9 : 1F
7338 CD E2 1F 0D 41 67 61 69 : 4D
7340 6E 20 28 59 2F 4E 29 20 : D5
7348 3F 20 00 CD 04 73 D8 CA : 45
7350 04 70 C3 56 70 21 6D 73 : FE
7358 0E 04 5E 23 56 23 06 20 : 32
7360 1A 13 BE 23 C0 10 F9 0D : E4
7368 20 F0 C9 00 00 10 80 20 : 89
7370 53 2D 4F 53 20 53 57 4F : 3B
7378 52 44 20 49 50 4C 20 73 : 2E
--------------------------------
SUM: B6 35 8D E2 08 28 95 8E 4BB6

7380 74 61 72 74 65 64 20 21 : C5
7388 0D 00 07 2A 2A 2A 2A 00 : BC
7390 9A ED 7B 6C 1F CD D6 1F : 4F
7398 3E 23 CD F4 1F ED 5B 76 : FF
73A0 1F CD D3 1F CD 1B 21 DC : C3
73A8 33 20 18 E5 1A FE 23 28 : B3
73B0 02 00 9F 7E 32 DE 27 EB : 41
73B8 29 29 29 29 EB E1 B7 28 : 4F
73C0 19 FE 80 30 19 3E 10 CD : FB
73C8 44 25 D8 11 00 10 19 E5 : 60
73D0 69 60 B7 80 A3 6F 63 61 : D6
73D8 74 69 6F 6E 20 54 61 62 : F1
73E0 6C 65 0D 46 69 6C 65 20 : 7E
73E8 6E 6F 74 20 46 6F 75 6E : 09
73F0 64 0D 44 65 76 01 8F 03 : 23
73F8 04 8F 00                : 93
--------------------------------
SUM: 52 E3 B7 A3 D2 0D F3 D3 E253
```

リスト8　システムジェネレータ

```
100 CLEAR ,&H1FFF : DEFINT A-Z : ON ERROR GOTO 440
110 LOADM "SYSGENo"          ' System genelate program
120 LOADM "SOSo"             ' SWORD-FM7 system module
130 COLOR 5 : PRINT "------- SWORD-FM7 system disk genelate program -------"
140 COLOR 7 : INPUT "Do you want Memory Editor in System Disk (Y/N) ?",IN$
150 IN=INSTR("YyNn",IN$) : IF IN=0 GOTO 140
160 IF IN<3 GOTO 220
170 '--- no Memory Editor
180 POKE &H21BC,40 ' IPL
190 POKE &H2021,40 ' SYSGENo
200 GOTO 240
210 '--- with Memory Editor
220 POKE &H21BC,43 ' IPL
230 POKE &H2021,43 ' SYSGENo
240 PRINT:INPUT "Drive Number=",DRV$
250 DRV=VAL(DRV$):IF DRV=0 AND DRV$<>"0" GOTO 240
260 IF DRV>3 GOTO 240
270 POKE &H2009,DRV
280 PRINT : PRINT "Mount phigical formatted disk on drive #";DRV
290 INPUT "Sure (Y/N) ? ",IN$
300 IN=INSTR("YyNn",IN$)
310 IF IN=0 GOTO 290 ELSE IF IN>2 THEN END
320 PRINT "Working."
330 EXEC &H2000
340 IF PEEK(&H2003)<>0 THEN ERROR 53
350 PRINT "Completed !"
360 INPUT "Genelate another system disk (Y/N) ? ",IN$
370 IN=INSTR("YyNn",IN$)
380 IF IN=0 GOTO 360 ELSE IF IN<3 GOTO 130
390 CLEAR ,&H71D5
400 END
410 '
420 ' Error trap routine
430 '
440 IF ERR<>53 THEN ON ERROR GOTO 0
450 COLOR 2
460 ERC=PEEK(&H2003)
470 IF ERC=10 THEN PRINT "Drive not ready." : GOTO 500
480 IF ERC=11 THEN PRINT "Disk write protected." : GOTO 500
490 PRINT "Device I/O error."
500 COLOR 7 : RESUME 360
```

リスト9　FAT，ディレクトリ作成

```
100 CLEAR 300,&H21FF
110 P=&H2200
120 READ A$
130 IF A$<>"*" THEN POKE P,VAL("&H"+A$):P=P+1:GOTO 120
140 FOR P=&H2208 TO &H224F
150     POKE P,0
160 NEXT
170 FOR P=&H2250 TO &H22FF
180     POKE P,&H8F
190 NEXT
200 FOR P=&H2300 TO &H33FF
210     POKE P,&HFF
220 NEXT
230 DATA 01,8F,03,04,8F,00,00,00,*
```

▶その昔，私はデゼニワールドを気長に待っていました。いま，私はNewBASICコンパイラをやはり気長に待っております。ねっ！　シャープさ〜ん，待ってるよー。

豊田　和彦（20）千葉県

リスト10 チェックサムプログラム

```
1000 '
1010 ' Check sum dump program
1020 '
1030 DEFINT A-Z : DIM VSUM(7)
1040 PRINT CHR$(&H11)+CHR$(5)+"├├├├├├├ Check sum dump program ├├├├├├├"
1050 INPUT "Start addr. =$",IN$
1060 SA=VAL("&H"+IN$)
1070 INPUT "End   addr. =$",IN$
1080 EA=VAL("&H"+IN$)
1090 INPUT "Dump to line printer (Y/N) ? ",IN$
1100 IF IN$="Y" OR IN$="y" THEN OPEN "O",#1,"LPT0:" ELSE OPEN "O",#1,"SCRN:
1110 FOR A=SA TO EA STEP 128
1120   FOR X=0 TO 7
1130     VSUM(X)=0
1140   NEXT
1150   FOR Y=0 TO 15
1160     HSUM=0
1170     PRINT #1,RIGHT$("000"+HEX$(A+Y*8),4)+" ";
1180     FOR X=0 TO 7
1190       D=PEEK(A+Y*8+X) : HSUM=HSUM+D : VSUM(X)=VSUM(X)+D
1200       PRINT #1,RIGHT$("0"+HEX$(D),2)+" ";
1210     NEXT
1220     PRINT #1,":"+RIGHT$("0"+HEX$(HSUM MOD 256),2)
1230   NEXT
1240   PRINT #1,"--------------------------------"
1250   PRINT #1,"Sum: ";
1260   TSUM=0
1270   FOR X=0 TO 7
1280     PRINT #1,RIGHT$("0"+HEX$(VSUM(X)MOD 256),2)+" "; : TSUM=TSUM+(VSUM(X) M
OD 256)
1290   NEXT
1300   PRINT #1,":"+RIGHT$("0"+HEX$(TSUM MOD 256),2)
1310   PRINT #1,""
1320 NEXT
1330 CLOSE #1
1340 END
```

リスト11 "SWORD"本体ソースリスト

```
0                  *
1                  *       S-OS 'SWORD-FM7' system module version 1.0
2                  *       Copyright 1985,86 by A.S.D.
3                  *       All rights reserved
4                  *
5                  *       Designe    : S.Kigoshi
6                  *       Code(6809) : S.Kigoshi
7                  *       Code(Z-80) : Y.Morishita
8                  *
9                  *       Hardware interface (Main)
10                 *
11        FC80     SHARED EQU   $FC80       Common memory

13        FD00     MTLPOT EQU   $FD00       Magnetic tape,printer control
14        FD01     LPDTOT EQU   $FD01       Line printer data out
15        FD02     MTLPIN EQU   $FD02       Magnetic tape,printer status
16        FD02     IRQCTL EQU   $FD02       Interupped request control
17        FD03     BUZZER EQU   $FD03       Buzzer control
18        FD04     STOPF  EQU   $FD04       'BREAK' key status
19        FD05     SBSIN  EQU   $FD05       Sub-system status
20        FD05     SBSOUT EQU   $FD05       Sub-system control
21        FD37     MLTPAG EQU   $FD37       Multi page control
22        FD38     COLPLT EQU   $FD38       Color palette control
23                 *
24                 *       Hardware interface (Sub)
25                 *
26        0000     __VB   EQU   $0000       VRAM (Blue)
27        4000     __VR   EQU   $4000       VRAM (Red)
28        8000     __VG   EQU   $8000       VRAM (Green)
29        C7D0     __ATB  EQU   $C7D0       Attr. buffer top addr.
30        D380     _COMAR EQU   $D380       Common memory

32        D409     VACC   EQU   $D409       VRAM access control
33        D40D     INSLED EQU   $D40D       INS-LED control
34        D40E     SOFFST EQU   $D40E       VRAM offset control
35                 *
36                 *       Constants
37                 *
38        0000     EOS    EQU   $00         End of string
39        0003     BRK    EQU   $03         Key-bord abort
40        0005     EL     EQU   $05         Erase line
41        0007     BEL    EQU   $07         Bell
42        0008     BS     EQU   $08         Back space
43        0009     HT     EQU   $09         Horizontal TAB
44        000A     LF     EQU   $0A         Line feed
45        000B     HOME   EQU   $0B         Cursor home
46        000C     EA     EQU   $0C         Erase all
47        000D     CR     EQU   $0D         Carriage return
48        0011     SF     EQU   $11         Set field
49        0012     SBA    EQU   $12         Set buffer addr.
50        0013     RC     EQU   $13         Repeat char.
51        001B     ESC    EQU   $1B         Escape
52        1B00     ESC00  EQU   $1B00       ESC*$100
53        001C     CSRR   EQU   $1C         Cursor right
54        001D     CSRL   EQU   $1D         Cursor left
55        001E     CSRU   EQU   $1E         Cursor up
56        001F     CSRD   EQU   $1F         Cursor down
57        0020     SPC    EQU   $20         Space
58        007F     DEL    EQU   $7F         Delete
59        00F1     PAI    EQU   $F1
60                 *
61                 *       Work area (Main)
62                 *
63        0000     ZRST00 EQU   $0000       Z-80 RST 00H GAP
64        0008     ZRST08 EQU   $0008       Z-80 RST 08H GAP
65        0010     ZRST10 EQU   $0010       Z-80 RST 10H GAP
66        0018     ZRST18 EQU   $0018       Z-80 RST 18H GAP
67        0020     ZRST20 EQU   $0020       Z-80 RST 20H GAP
68        0028     ZRST28 EQU   $0028       Z-80 RST 28H GAP
69        0030     ZRST30 EQU   $0030       Z-80 RST 30H GAP
70        0038     ZRST38 EQU   $0038       Z-80 RST 38H GAP
71        0066     ZNMI66 EQU   $0066       Z-80 NMI     GAP

73        0080     ZSTKPT EQU   $0080       Z-80 System stack pointer save
74        0082     FUNCNO EQU   $0082       System call function nunber
75        0083     CNTCHR EQU   $0083       Char. counter
76        0084     CSRX   EQU   $0084       Cursor X pos.
77        0085     CSRY   EQU   $0085       Cursor Y pos.
78        0086     ILIMIT EQU   $0086       Buffer limit (Input 1 line)
79        0087     BRKFLG EQU   $0087       Break key flag
80        0088     CRATTR EQU   $0088       Current attr.
81        0089     CRCNSM EQU   $0089       Current console mode
82        008A     LPRTRY EQU   $008A       Line printer retry count (Sec)
83        008B     MTLPWK EQU   $008B       I/O $FD00 write work
84        008C     IRQWRK EQU   $008C       I/O $FD02 write work
85        008D     BELWRK EQU   $008D       I/O $FD03 write work
86        008E     SBSWRK EQU   $008E       I/O $FD05 write work
87        008F     MPWRK  EQU   $008F       Multi page control work

89        0400     Z80STK EQU   $0400       Z-80 system stack bottom
90        0600     STKBTM EQU   $0600       6809 system stack bottom

92        0600     SINBUF EQU   $0600       System input buffer
93        0700     INFBLK EQU   $0700       Information block
94        0701     INFBK1 EQU   INFBLK+1
95        0701     NAME@@ EQU   $0701       File name
96        0712     SIZE@@ EQU   $0712       File size
97        0714     DTADR@ EQU   $0714       Data addr.
98        0716     EXADR@ EQU   $0716       Exec addr.
99                 *
100                *       Work area (Sub)
101                *
102       D01F     _VOFST EQU   $D01F       VRAM offset work
103       D022     PUTCFG EQU   $D022       Continuance PUT flag
104       D054     GTTOPA EQU   $D054       Current window attr. top addr.
105       D056     GTBTMA EQU   $D056       Current window attr. bottom addr
106       D05C     _CAAD  EQU   $D05C       Cursor attr. buffer addr.
107                *
108                *       System sub-routine (Main)
109                *
110       FE05     DWRITE EQU   $FE05       5" disk 1 sector write routine
111       FE08     DREAD  EQU   $FE08       5" disk 1 sector read rotuine
112                *
113                *       System sub-routine (Sub)
114                *
115       E3C3     CBADR  EQU   $E3C3       Calculate buffer address
116       E787     CTOB   EQU   $E787       Set buffer pointer from cursor
117       EA7F     GETENA EQU   $EA7F       Set cursor

119                *
120                * +++  SWORD 6809 system module  +++
121                *       Coded by S.Kigoshi
122                *
123 0800                  ORG   $0800
124       00              SETDP $00
125                *
126                *       System initialize routine entry
127                *
128 0800   20 27   ENTRY  BRA   SYSINT
129                *
130                *       System module ID code
131                *
132 0802  53              FCC   'SOSSYSV1'
133 080A  00              FCB   EOS
134                *
135                *       6809 interupped vector data
136                *
137 080B  0880     IRQVCT FDB   HKSWI3      SWI3 vector
138 080D  0883            FDB   HKSWI2      SWI2 vector
139 080F  0886            FDB   HKFIRQ      FIRQ vector
140 0811  0889            FDB   HKIRQ       IRQ  vector
141 0813  088C            FDB   HKSWI       SWI  vector
142 0815  088F            FDB   HKNMI       NMI  vector
143                *
144                *       Z-80 interupped vector data
145                *
146 0817  00       ZIVTBL FCB   <ZHK00H,>ZHK00H ' For RST 00H
147 0819  03              FCB   <ZHK08H,>ZHK08H ' For RST 08H
148 081B  06              FCB   <ZHK10H,>ZHK10H ' For RST 10H
149 081D  09              FCB   <ZHK18H,>ZHK18H ' For RST 18H
150 081F  0C              FCB   <ZHK20H,>ZHK20H ' For RST 20H
151 0821  0F              FCB   <ZHK28H,>ZHK28H ' For RST 28H
152 0823  12              FCB   <ZHK30H,>ZHK30H ' For RST 30H
153 0825  15              FCB   <ZHK38H,>ZHK38H ' For RST 38H
154 0827  18              FCB   <ZHK66H,>ZHK66H ' For NMI
155                *
156                *       System initialize routine
157                *
158 0829   1A 50   SYSINT ORCC  #%01010000
159 082B   4F             CLRA
160 082C   1F 8B          TFR   A,DP        Set-up DP reg.
161 082E 10CE 0600        LDS   #STKBTM     Set-up system stack
162                *
163                *       6809 interupped vector initialize
164                *
165 0832   30 8CD6        LEAX  IRQVCT,PCR Initialize data top
```

▶ モアイのばか！（グラディウスのことです） 大坪 建治（18）香川県

```
166 Ø835   CE FFF2          LDU   #$FFF2     Interupped vector top
167 Ø838   C6 ØC            LDB   #12        Table length
168 Ø83A   8D 7D            BSR   BLKMV1     Set data
169                  *
170                  *      Z-8Ø interupped hook initialize
171                  *
172 Ø83C   31 8CD8          LEAY  ZIVTBL,PCR
173 Ø83F   CE ØØØØ          LDU   #ZRSTØØ
174 Ø842   CC C3Ø8          LDD   #$C3Ø8
175 Ø845   A7 C4     ZINTLP STA   ,U
176 Ø847   AE A1            LDX   ,Y++
177 Ø849   AF 41            STX   1,U
178 Ø84B   33 48            LEAU  8,U
179 Ø84D   5A               DECB
18Ø Ø84E   26 F5            BNE   ZINTLP
181 Ø85Ø   33 C836          LEAU  $66-$38+8,U
182 Ø853   A7 C4            STA   ,U
183 Ø855   AE A4            LDX   ,Y
184 Ø857   AF 41            STX   1,U
185                  *
186                  *      Initialize I/O port
187                  *
188 Ø859   86 4Ø            LDA   #%Ø1ØØØØØØ
189 Ø85B   97 8B            STA   MTLPWK
19Ø Ø85D   B7 FDØØ          STA   MTLPOT     Init. "LPTØ:" , "CASØ:"
191 Ø86Ø   4F               CLRA
192 Ø861   97 8C            STA   IRQWRK
193 Ø863   B7 FDØ2          STA   IRQCTL     Mask all IRQ
194 Ø866   97 8E            STA   SBSWRK
195 Ø868   4C               INCA
196 Ø869   97 8D            STA   BELWRK
197 Ø86B   B7 FDØ3          STA   BUZZER     Buzzer enable
198                  *
199                  *      Jump to command handler
2ØØ                  *
2Ø1 Ø86E   86 14            LDA   #2Ø        2Ø sec.
2Ø2 Ø87Ø   97 8A            STA   LPRTRY
2Ø3 Ø872   86 FF            LDA   #255       255 char.
2Ø4 Ø874   97 86            STA   ILIMIT
2Ø5 Ø876   1C AF            ANDCC #%1Ø1Ø1111 Enable interupped
2Ø6 Ø878   16 Ø7B8          LBRA  CMDENT

2Ø8 Ø87B      ØØØ5          RMB   $Ø88Ø-*
2Ø9                  *
21Ø                  *      68Ø9 interupped hook
211                  *
212 Ø88Ø   7E Ø892   HKSWI3 JMP   IRQDMY
213 Ø883   7E Ø892   HKSWI2 JMP   IRQDMY
214 Ø886   7E Ø893   HKFIRQ JMP   IRQBRK
215 Ø889   7E Ø892   HKIRQ  JMP   IRQDMY
216 Ø88C   7E Ø892   HKSWI  JMP   IRQDMY
217 Ø88F   7E Ø892   HKNMI  JMP   IRQDMY
218                  *
219                  *      IRQ dummy return
22Ø                  *
221 Ø892   3B        IRQDMY RTI
222                  *
223                  *      FIRQ handler
224                  *
225 Ø893   34 Ø2     IRQBRK PSHS  A
226 Ø895   B6 FDØ4          LDA   STOPF
227 Ø898   84 Ø2            ANDA  #%ØØØØØØ1Ø Is 'BREAK' key pressed ?
228 Ø89A   26 1A            BNE   IBRKEX     No. RTI
229 Ø89C   43               COMA
23Ø Ø89D   97 87            STA   BRKFLG     Key-board abort flag on
231 Ø89F   96 8E            LDA   SBSWRK
232 Ø8A1   2B ØC            BMI   IBRKLP     Don't atention,when sub is halt
233 Ø8A3   8A 4Ø            ORA   #%Ø1ØØØØØØ
234 Ø8A5   B7 FDØ5          STA   SBSOUT     Atention interupped request
235 Ø8A8   84 BF            ANDA  #%1Ø111111
236 Ø8AA   B7 FDØ5          STA   SBSOUT     Remove atention IRQ
237 Ø8AD   97 8E            STA   SBSWRK
238 Ø8AF   B6 FDØ4   IBRKLP LDA   STOPF
239 Ø8B2   85 Ø2            BITA  #%ØØØØØØ1Ø
24Ø Ø8B4   27 F9            BEQ   IBRKLP     Wait 'BREAK' key open
241 Ø8B6   35 Ø2     IBRKEX PULS  A
242 Ø8B8   3B               RTI
243                  *
244                  *      System sub-routine package
245                  *

247                  *
248                  *      Block transfer 1 (<256 bytes)
249                  *
25Ø                  *      IN     U:Dest. top
251                  *             X:Src. top
252                  *             B:Src. len
253                  *      OUT    U:Next dest. top
254                  *             X:Next src. top
255                  *             B:Ø
256                  *      BRK    A
257                  *
258 Ø8B9   A6 8Ø     BLKMV1 LDA   ,X+
259 Ø8BB   A7 CØ            STA   ,U+
26Ø Ø8BD   5A               DECB
261 Ø8BE   26 F9            BNE   BLKMV1
262 Ø8CØ   39               RTS
263                  *
264                  *      Block transfer 2
265                  *
266                  *      IN     U:Dest. top
267                  *             Y:Src. len
268                  *             X:Src. top
269                  *      OUT    U:Next dest. top
27Ø                  *             Y:Ø
271                  *             X:Next src. top
272                  *      BRK    A
273                  *
274 Ø8C1   A6 8Ø     BLKMV2 LDA   ,X+
275 Ø8C3   A7 CØ            STA   ,U+
276 Ø8C5   31 3F            LEAY  -1,Y
277 Ø8C7   26 F8            BNE   BLKMV2
278 Ø8C9   39               RTS
279                  *
28Ø                  *      Sub-system halt
281                  *
282                  *      IN     /
283                  *      OUT    /
284                  *      BRK    /
285                  *
286 Ø8CA   34 Ø2     SUBHLT PSHS  A
287 Ø8CC   B6 FDØ5   SUBHL1 LDA   SBSIN      Is sub-system ready ?
288 Ø8CF   2B FB            BMI   SUBHL1     No. wait
289 Ø8D1   96 8E            LDA   SBSWRK
29Ø Ø8D3   8A 8Ø            ORA   #%1ØØØØØØØ
291 Ø8D5   97 8E            STA   SBSWRK
292 Ø8D7   B7 FDØ5          STA   SBSOUT     Sub-system halt
293 Ø8DA   B6 FDØ5   SUBHL2 LDA   SBSIN      Is sub-system halt sure ?
294 Ø8DD   2A FB            BPL   SUBHL2     No. wait
295 Ø8DF   35 82            PULS  PC,A       Return
296                  *
297                  *      Sub-system restart/ready request
298                  *
299                  *      IN     /
3ØØ                  *      OUT    /
3Ø1                  *      BRK    /
3Ø2                  *
3Ø3 Ø8E1   34 Ø2     RDYREQ PSHS  A
3Ø4 Ø8E3   B6 FC8Ø          LDA   SHARED
3Ø5 Ø8E6   8A 8Ø            ORA   #%1ØØØØØØØ
3Ø6 Ø8E8   B7 FC8Ø          STA   SHARED
3Ø7 Ø8EB      8C            FCB   $8C        CMPX #
3Ø8 Ø8EC   34 Ø2     SUBRST PSHS  A
3Ø9 Ø8EE   96 8E            LDA   SBSWRK
31Ø Ø8FØ   84 7F            ANDA  #%Ø1111111
311 Ø8F2   B7 FDØ5          STA   SBSOUT     Restart sub-system
312 Ø8F5   97 8E            STA   SBSWRK
313 Ø8F7   35 82            PULS  PC,A
314                  *
315                  * +++  I/O handler  +++
316                  *

318                  *
319                  *      Dummy GAP for command error
32Ø                  *
321 Ø8F9   39        CMDUMY RTS
322                  *
323                  *      Get cursor location
324                  *
325 Ø8FA   CE FC82   GET_XY LDU   #SHARED+2
326 Ø8FD   86 ØA            LDA   #$ØA       GBADR
327 Ø8FF   8D C9            BSR   SUBHLT
328 Ø9Ø1   A7 C4            STA   ,U
329 Ø9Ø3   8D E7            BSR   SUBRST
33Ø Ø9Ø5   8D C3            BSR   SUBHLT
331 Ø9Ø7   EC 41            LDD   1,U
332 Ø9Ø9   DD 84            STD   CSRX
333 Ø9ØB   2Ø D4            BRA   RDYREQ
334                  *
335                  *      Set cursor location
336                  *
337 Ø9ØD   34 46     SET_XY PSHS  U,A,B
338 Ø9ØF   CE FC81          LDU   #SHARED+1
339 Ø912   CC Ø3Ø3          LDD   #$Ø3Ø3     PUT
34Ø Ø915   8D B3            BSR   SUBHLT
341 Ø917   6F CØ            CLR   ,U+
342 Ø919   ED C1            STD   ,U++
343 Ø91B   86 12            LDA   #SBA
344 Ø91D   A7 CØ            STA   ,U+
345 Ø91F   DC 84            LDD   CSRX
346 Ø921   ED C4            STD   ,U
347 Ø923   35 46            PULS  U,A,B
348 Ø925   2Ø C5            BRA   SUBRST
349                  *
35Ø                  *      Display cursor
351                  *
352 Ø927   34 Ø6     DSPCUR PSHS  A,B
353 Ø929   D6 89            LDB   CRCNSM     Crrent console mode
354 Ø92B   CA Ø1            ORB   #%ØØØØØØØ1 Set cursor display flag.
355 Ø92D   2Ø Ø6            BRA   CLRCU1
356                  *
357                  *      Clear cursor
358                  *
359 Ø92F   34 Ø6     CLRCUR PSHS  A,B
36Ø Ø931   D6 89            LDB   CRCNSM     Crrent console mode
361 Ø933   C4 3E            ANDB  #%ØØ11111Ø Remove cursor display flag.
362 Ø935   D7 89     CLRCU1 STB   CRCNSM
363 Ø937   86 ØC            LDA   #$ØC       CNSCTL
364 Ø939   8D 8F            BSR   SUBHLT
365 Ø93B   FD FC82          STD   SHARED+2
366 Ø93E   8D AC            BSR   SUBRST
367 Ø94Ø   35 86            PULS  PC,A,B
368                  *
369                  *      Flash key buffer
37Ø                  *
371 Ø942   34 46     FLSBUF PSHS  U,A,B
372 Ø944   CE FC81          LDU   #SHARED+1
373 Ø947   CC Ø3Ø2          LDD   #$Ø3Ø2     PUT
374 Ø94A   17 FF7D          LBSR  SUBHLT
375 Ø94D   6F CØ            CLR   ,U+
376 Ø94F   ED C1            STD   ,U++
377 Ø951   CC 1B39          LDD   #ESCØØ+'9
378 Ø954   ED C4            STD   ,U
379 Ø956   35 46            PULS  U,A,B
38Ø Ø958   2Ø 92            BRA   SUBRST
381                  *
382                  *      Console initialize
383                  *
384 Ø95A   EC A81Ø   INIT   LDD   16,Y       Initialize parameter (HL)
385 Ø95D   1E 89            EXG   A,B
386 Ø95F   1F Ø1            TFR   D,X
387 Ø961   CE FC82          LDU   #SHARED+2
388 Ø964   CC Ø1Ø8          LDD   #$Ø1Ø8     INIT
389 Ø967   17 FF6Ø          LBSR  SUBHLT
39Ø Ø96A   A7 CØ            STA   ,U+
391 Ø96C   17 FF4A          LBSR  BLKMV1
392 Ø96F   17 FF7A          LBSR  SUBRST
393 Ø972   2Ø 86            BRA   GET_XY
394                  *
395                  *      Console mode control
396                  *
397 Ø974   CE FC82   CNSCTL LDU   #SHARED+2
398 Ø977   86 ØC            LDA   #$ØC       CNSCTL
399 Ø979   E6 2B            LDB   11,Y       Console mode parameter (A)
4ØØ Ø97B   C4 3F            ANDB  #%ØØ111111
4Ø1 Ø97D   D7 89            STB   CRCNSM     Crrent console mode
4Ø2 Ø97F   17 FF48          LBSR  SUBHLT
4Ø3 Ø982   ED C4            STD   ,U
4Ø4 Ø984   16 FF65          LBRA  SUBRST
4Ø5                  *
4Ø6                  *      Set console TAB location
4Ø7                  *
4Ø8 Ø987   EC A81Ø   TABSET LDD   16,Y       TAB table top addr. (HL)
4Ø9 Ø98A   1E 89            EXG   A,B
41Ø Ø98C   1F Ø1            TFR   D,X
411 Ø98E   CC ØBØA          LDD   #$ØBØA     TABSET
412 Ø991   CE FC82          LDU   #SHARED+2
413 Ø994   17 FF33          LBSR  SUBHLT
414 Ø997   A7 CØ            STA   ,U+
415 Ø999   17 FF1D          LBSR  BLKMV1
416 Ø99C   16 FF4D          LBRA  SUBRST
417                  *
418                  *      Set console attribute code
419                  *
42Ø Ø99F   A6 2B     SETATR LDA   11,Y       Attribute code (A)
421 Ø9A1   84 1F            ANDA  #%ØØØ11111
422 Ø9A3   97 88            STA   CRATTR
423 Ø9A5   39               RTS
424                  *
425                  *      Multi page control
426                  *
427 Ø9A6   B6 FDØ5   MPCTL  LDA   SBSIN
428 Ø9A9   2B FB            BMI   MPCTL
429 Ø9AB   A6 2B            LDA   11,Y       Multi page parameter (A)
43Ø Ø9AD   84 77            ANDA  #%Ø111Ø111
431 Ø9AF   97 8F            STA   MPWRK
432 Ø9B1   43               COMA
433 Ø9B2   B7 FD37          STA   MLTPAG
434 Ø9B5   39               RTS
435                  *
436                  *      Color palette control
437                  *
438 Ø9B6   8E FD38   CPCTL  LDX   #COLPLT
439 Ø9B9   A6 2B            LDA   11,Y       Coror palette parameter (A)
44Ø Ø9BB   E6 2B            LDB   11,Y       Coror palette parameter (A)
441 Ø9BD   84 Ø7            ANDA  #%ØØØØØ111
442 Ø9BF   54               LSRB
443 Ø9CØ   54               LSRB
444 Ø9C1   54               LSRB
445 Ø9C2   54               LSRB
446 Ø9C3   C4 Ø7            ANDB  #%ØØØØØ111
447 Ø9C5   A7 85            STA   B,X
448 Ø9C7   39               RTS
449                  *
45Ø                  *      Input 1 charactor from key-board with wait
451                  *
452 Ø9C8   17 FF64   INCH   LBSR  CLRCUR     Clear cursor
453 Ø9CB   17 FF3F          LBSR  SET_XY     Set cursor location
454 Ø9CE   17 FF56          LBSR  DSPCUR     Display cursor
455 Ø9D1   8D ØD     INCHLP BSR   INCHR1
456 Ø9D3   4D               TSTA
457 Ø9D4   27 FB            BEQ   INCHLP
458 Ø9D6   39               RTS
459                  *
```

```
460                      *      Input 1 charactor from key-board real time
461                      *
462 09D7    17 FF55    INCHR  LBSR   CLRCUR      Clear cursor
463 09DA    17 FF30           LBSR   SET_XY      Set cursor location
464 09DD    17 FF47           LBSR   DSPCUR      Display cursor
465 09E0    96 87      INCHR1 LDA    BRKFLG      Break ?
466 09E2    27 09             BEQ    INCHR2      No,
467 09E4    0F 87             CLR    BRKFLG
468 09E6    86 1B             LDA    #ESC
469 09E8    A7 2B             STA    11,Y        Key code (A)
470 09EA    16 FF55           LBRA   FLSBUF

472 09ED    CE FC82    INCHR2 LDU    #SHARED+2
473 09F0    CC 2900           LDD    #$2900      INKEY
474 09F3    17 FED4           LBSR   SUBHLT
475 09F6    ED C4             STD    ,U
476 09F8    17 FEF1           LBSR   SUBRST
477 09FB    17 FECC           LBSR   SUBHLT
478 09FE    EC 41             LDD    1,U
479 0A00    17 FEDE           LBSR   RDYREQ
480 0A03    5D                TSTB
481 0A04    27 0D             BEQ    INCHR3
482 0A06    81 87             CMPA   #'■
483 0A08    26 02             BNE    INCHRA
484 0A0A    86 7B             LDA    #'{
485 0A0C    81 F1      INCHRA CMPA   #PAI
486 0A0E    26 04             BNE    INCHR4
487 0A10    86 7F             LDA    #DEL
488 0A12       21             FCB    $21         BRN
489 0A13    4F         INCHR3 CLRA
490 0A14    A7 2B      INCHR4 STA    11,Y        Key code (A)
491 0A16    39                RTS
492                      *
493                      *      Check 'BREAK' key status
494                      *
495 0A17    A6 2A      CHKBRK LDA    10,Y        (F)
496 0A19    D6 87             LDB    BRKFLG
497 0A1B    27 06             BEQ    CHKBR1
498 0A1D    17 FF22           LBSR   FLSBUF
499 0A20    8A 40             ORA    #%01000000
500 0A22       8C             FCB    $8C         CMPX #
501 0A23    84 BF      CHKBR1 ANDA   #%10111111
502 0A25    A7 2A             STA    10,Y        Break key flag (F)
503 0A27    0F 87             CLR    BRKFLG
504 0A29    39                RTS
505                      *
506                      *      Output chractor to display
507                      *
508 0A2A    A6 2B      OUTCH  LDA    11,Y        Put char. code (A)
509 0A2C    34 02      OUTCH0 PSHS   A
510 0A2E    17 FEFE           LBSR   CLRCUR      Clear cursor
511 0A31    17 FED9           LBSR   SET_XY      Set cursor location
512 0A34    17 FEF0           LBSR   DSPCUR      Display cursor
513 0A37    0C 83             INC    CNTCHR
514 0A39    CE FC81           LDU    #SHARED+1
515 0A3C    CC 0301           LDD    #$0301      PUT
516 0A3F    17 FE88           LBSR   SUBHLT
517 0A42    6F C0             CLR    ,U+
518 0A44    ED C1             STD    ,U++
519 0A46    35 02             PULS   A
520 0A48    81 7B             CMPA   #'{
521 0A4A    26 02             BNE    OUTCHA
522 0A4C    86 87             LDA    #'■
523 0A4E    81 7F      OUTCHA CMPA   #DEL
524 0A50    26 02             BNE    OUTCH1
525 0A52    86 F1             LDA    #PAI
526 0A54    A7 C4      OUTCH1 STA    ,U
527 0A56    81 0D             CMPA   #CR
528 0A58    26 0D             BNE    OUTCH2
529 0A5A    0F 83             CLR    CNTCHR
530 0A5C    86 11             LDA    #SF
531 0A5E    D6 88             LDB    CRATTR
532 0A60    ED 41             STD    1,U
533 0A62    86 03             LDA    #3
534 0A64    B7 FC83           STA    SHARED+3
535 0A67    17 FE82    OUTCH2 LBSR   SUBRST
536 0A6A    16 FE8D           LBRA   GET_XY

538            D0             SETDP  $D0
539                      *
540                      *      Sub-system test command 1
541                      *Remove all modification flag & set it on cursor loc.
542                      *
543 0A6D       00      TST1   FCB    0,0,$3F
544 0A70       59             FCC    'YAMAUCHI'
545 0A78       93             FCB    $93
546 0A79       D38F           FDB    TST11-TST1+_COMAR
547 0A7B       90             FCB    $90

549 0A7C    9E 54      TST11  LDX    GTTOPA
550 0A7E    A6 84      TST12  LDA    ,X
551 0A80    84 BF             ANDA   #%10111111
552 0A82    A7 80             STA    ,X+
553 0A84    9C 56             CMPX   GTBTMA
554 0A86    25 F6             BCS    TST12
555 0A88    9E 5C             LDX    _CAAD
556 0A8A    A6 84             LDA    ,X
557 0A8C    8A 40             ORA    #%01000000
558 0A8E    A7 84             STA    ,X
559 0A90    30 01      TST13  LEAX   1,X
560 0A92    A6 84             LDA    ,X
561 0A94    2B 04             BMI    TST14
562 0A96    9C 56             CMPX   GTBTMA
563 0A98    23 F6             BLS    TST13
564 0A9A    30 1F      TST14  LEAX   -1,X
565 0A9C    A6 89F830         LDA    -2000,X
566 0AA0    26 04             BNE    TST15
567 0AA2    A6 84             LDA    ,X
568 0AA4    2A F4             BPL    TST14
569 0AA6    1F 12      TST15  TFR    X,Y
570 0AA8    BD EA7F           JSR    GETENA
571 0AAB    BD E3C3           JSR    CBADR
572 0AAE    7E E787           JMP    CTOB

574            0AB1    TST1EN EQU    *

576            00             SETDP  $00
577                      *
578                      *      Input 1 line from console
579                      *
580 0AB1    17 FE7B    INLN   LBSR   CLRCUR      Clear cursor
581 0AB4    17 FE56           LBSR   SET_XY      Set cursor location
582 0AB7    17 FE6D           LBSR   DSPCUR      Display cursor
583 0ABA    0F 87             CLR    BRKFLG      Assume no break
584 0ABC    EC 2E             LDD    14,Y        Get input buffer addr.(DE)
585 0ABE    1E 89             EXG    A,B
586 0AC0    1F 01             TFR    D,X
587 0AC2    CC 0000           LDD    #$0000
588 0AC5    A7 80      CLRBUF STA    ,X+
589 0AC7    5A                DECB
590 0AC8    26 FB             BNE    CLRBUF
591 0ACA    CE FC81    INLN1  LDU    #SHARED+1
592 0ACD    CC 0400           LDD    #$0400      GET
593 0AD0    17 FDF7           LBSR   SUBHLT
594 0AD3    6F C0             CLR    ,U+
595 0AD5    ED C4             STD    ,U
596 0AD7    17 FE12           LBSR   SUBRST
597 0ADA    30 8C90           LEAX   TST1,PCR
598 0ADD    CE FC80           LDU    #SHARED
599 0AE0    17 FDE7           LBSR   SUBHLT
600 0AE3    A6 43             LDA    3,U         Get end code
601 0AE5    E6 41      INLNC  LDB    1,U         Continue ?
602 0AE7    2A 0E             BPL    INLNB       No,
603 0AE9    6F 41             CLR    1,U
604 0AEB    C6 64             LDB    #$64        CONT
605 0AED    E7 42             STB    2,U
606 0AEF    17 FDFA           LBSR   SUBRST
607 0AF2    17 FDD5           LBSR   SUBHLT
608 0AF5    20 EE             BRA    INLNC

610 0AF7    4D         INLNB  TSTA
611 0AF8    27 0D             BEQ    INLN2
612 0AFA    17 FDE4           LBSR   RDYREQ
613 0AFD    4A                DECA               CTRL-C ?
614 0AFE    26 CA             BNE    INLN1       No. retry
615 0B00    86 FF             LDA    #-1
616 0B02    97 87             STA    BRKFLG      Set break flag
617 0B04    17 FDC3           LBSR   SUBHLT
618 0B07    C6 44      INLN2  LDB    #TST1EN-TST1
619 0B09    17 FDAD           LBSR   BLKMV1
620 0B0C    17 FDDD           LBSR   SUBRST
621 0B0F    CE FC82           LDU    #SHARED+2
622 0B12    86 05             LDA    #$05        GETC
623 0B14    17 FDB3           LBSR   SUBHLT
624 0B17    A7 C1             STA    ,U++        !
625 0B19    17 FDD0           LBSR   SUBRST
626 0B1C    EC 2E             LDD    14,Y        Input buffer addr. (DE)
627 0B1E    1E 89             EXG    A,B
628 0B20    1F 01             TFR    D,X
629 0B22    D6 86             LDB    ILIMIT
630 0B24    34 04             PSHS   B
631 0B26    17 FDA1           LBSR   SUBHLT
632 0B29    E6 C0             LDB    ,U+
633 0B2B    C0 03             SUBB   #3
634 0B2D    23 35             BLS    INLN6
635 0B2F    33 43             LEAU   3,U
636 0B31    A6 C0      INLN3  LDA    ,U+
637 0B33    26 02             BNE    INLN4       Null -> SPC
638 0B35    86 20             LDA    #'          Space
639 0B37    81 87      INLN4  CMPA   #'■
640 0B39    26 02             BNE    INLNA
641 0B3B    86 7B             LDA    #'{
642 0B3D    81 F1      INLNA  CMPA   #PAI
643 0B3F    26 02             BNE    INLN5
644 0B41    86 7F             LDA    #DEL
645 0B43    A7 80      INLN5  STA    ,X+
646 0B45    6A E4             DEC    ,S
647 0B47    27 1B             BEQ    INLN6
648 0B49    5A                DECB
649 0B4A    26 E5             BNE    INLN3
650 0B4C    B6 FC81           LDA    SHARED+1
651 0B4F    2A 13             BPL    INLN6
652 0B51    CE FC81           LDU    #SHARED+1
653 0B54    86 64             LDA    #$64        CONT
654 0B56    6F C0             CLR    ,U+
655 0B58    A7 C1             STA    ,U++
656 0B5A    17 FD8F           LBSR   SUBRST
657 0B5D    17 FD6A           LBSR   SUBHLT
658 0B60    E6 C0             LDB    ,U+
659 0B62    26 CD             BNE    INLN3
660 0B64    17 FD7A    INLN6  LBSR   RDYREQ
661 0B67    32 61             LEAS   1,S
662 0B69    17 FD8E           LBSR   GET_XY
663 0B6C    6F 84             CLR    ,X          Set end code
664 0B6E    86 0D             LDA    #CR
665 0B70    17 FEB9           LBSR   OUTCH0      Move cursor to next pos.
666 0B73    EC 2E             LDD    14,Y        Input buffer addr. (DE)
667 0B75    1E 89             EXG    A,B
668 0B77    1F 01             TFR    D,X
669 0B79    96 87             LDA    BRKFLG
670 0B7B    27 07             BEQ    INLN7
671 0B7D    CC 1B00           LDD    #ESC00
672 0B80    ED 84             STD    ,X
673 0B82    D7 87             STB    BRKFLG
674 0B84    39         INLN7  RTS
675                      *
676                      *      Input multi line from console
677                      *
678 0B85    17 FDA7    INLNM  LBSR   CLRCUR      Clear cursor
679 0B88    17 FD82           LBSR   SET_XY      Set cursor location
680 0B8B    17 FD99           LBSR   DSPCUR      Display cursor
681 0B8E    0F 87             CLR    BRKFLG
682 0B90    CE FC82    INLNM1 LDU    #SHARED+2
683 0B93    86 05             LDA    #$05        GETC
684 0B95    17 FD32           LBSR   SUBHLT
685 0B98    A7 C0      INLNMA STA    ,U+
686 0B9A    17 FD4F           LBSR   SUBRST
687 0B9D    17 FD2A           LBSR   SUBHLT
688 0BA0    96 87             LDA    BRKFLG
689 0BA2    26 0A             BNE    INLNM2      Abort
690 0BA4    A6 C0             LDA    ,U+         End code is CR
691 0BA6    27 15             BEQ    INLNM3      Yes CR
692 0BA8    17 FD36           LBSR   RDYREQ
693 0BAB    4A                DECA               Abort ?
694 0BAC    26 E2             BNE    INLNM1      No. retry
695 0BAE    0F 87      INLNM2 CLR    BRKFLG
696 0BB0    EC 2E             LDD    14,Y        Input buffer addr. (DE)
697 0BB2    1E 89             EXG    A,B
698 0BB4    1F 01             TFR    D,X
699 0BB6    86 1B             LDA    #ESC
700 0BB8    A7 84             STA    ,X
701 0BBA    16 FD3D           LBRA   GET_XY

703 0BBD    D6 86      INLNM3 LDB    ILIMIT
704 0BBF    34 04             PSHS   B
705 0BC1    EC 2E             LDD    14,Y        Inout buffer addr. (DE)
706 0BC3    1E 89             EXG    A,B
707 0BC5    1F 01             TFR    D,X
708 0BC7    E6 C4             LDB    ,U
709 0BC9    33 43             LEAU   3,U
710 0BCB    C0 03             SUBB   #3
711 0BCD    23 3D             BLS    INLNM8      No input
712 0BCF    A6 C0      INLNM4 LDA    ,U+
713 0BD1    26 02             BNE    INLNM5      Null -> SPC
714 0BD3    86 20             LDA    #'          Space
715 0BD5    81 87      INLNM5 CMPA   #'■
716 0BD7    26 02             BNE    INLNMB
717 0BD9    86 7B             LDA    #'{
718 0BDB    81 F1      INLNMB CMPA   #PAI
719 0BDD    26 02             BNE    INLNM6
720 0BDF    86 7F             LDA    #DEL
721 0BE1    A7 80      INLNM6 STA    ,X+
722 0BE3    6A E4             DEC    ,S
723 0BE5    27 1B             BEQ    INLNM7
724 0BE7    5A                DECB
725 0BE8    26 E5             BNE    INLNM4
726 0BEA    B6 FC81           LDA    SHARED+1
727 0BED    2A 13             BPL    INLNM7
728 0BEF    CE FC81           LDU    #SHARED+1
729 0BF2    86 64             LDA    #$64        Cont
730 0BF4    6F C0             CLR    ,U+
731 0BF6    A7 C1             STA    ,U++        !
732 0BF8    17 FCF1           LBSR   SUBRST
733 0BFB    17 FCCC           LBSR   SUBHLT
734 0BFE    E6 C0             LDB    ,U+
735 0C00    26 CD             BNE    INLNM4
736 0C02    17 FCDC    INLNM7 LBSR   RDYREQ
737 0C05    32 61             LEAS   1,S
738 0C07    6F 84             CLR    ,X
739 0C09    16 FCEE           LBRA   GET_XY

741 0C0C    CE FC82    INLNM8 LDU    #SHARED+2
742 0C0F    86 03             LDA    #$03
743 0C11    7F FC83           CLR    SHARED+3
744 0C14    20 82             BRA    INLNMA
745                      *
746                      *      Output 1 line to console
747                      *
748 0C16    17 FD16    OUTLN  LBSR   CLRCUR      Clear cursor
749 0C19    17 FCF1           LBSR   SET_XY      Set cursor location
750 0C1C    17 FD08           LBSR   DSPCUR      Display cursor
751 0C1F    EC 2E             LDD    14,Y        Output buffer addr. (DE)
752 0C21    1E 89             EXG    A,B
753 0C23    1F 01             TFR    D,X
```

▶ MZ-2861の紹介記事の「終わりに」は，非常に的確に現状をとらえており，ちょうちん記事に終わらず好感が持てた。　　水野　正克（22）愛知県

```
 754 0C25    A6 2B           LDA   11,Y        Delimitor (A)
 755 0C27    34 02    OUTLN0 PSHS  A
 756 0C29    CE FC81  OUTLN1 LDU   #SHARED+1
 757 0C2C    CC 0300         LDD   #$0300      PUT
 758 0C2F    17 FC98         LBSR  SUBHLT
 759 0C32    6F C0           CLR   ,U+
 760 0C34    A7 C1           STA   ,U++        !
 761 0C36    A6 80    OUTLN2 LDA   ,X+
 762 0C38    A1 E4           CMPA  ,S
 763 0C3A    26 0B           BNE   OUTLN3
 764 0C3C    F7 FC83         STB   SHARED+3
 765 0C3F    17 FCAA         LBSR  SUBRST
 766 0C42    32 61           LEAS  1,S
 767 0C44    16 FCB3         LBRA  GET_XY

 769 0C47    0C 83    OUTLN3 INC   CNTCHR
 770 0C49    81 7B           CMPA  #'(
 771 0C4B    26 02           BNE   OUTLNA
 772 0C4D    86 87           LDA   #'■
 773 0C4F    81 7F    OUTLNA CMPA  #DEL
 774 0C51    26 02           BNE   OUTLN4
 775 0C53    86 F1           LDA   #PAI
 776 0C55    A7 C0    OUTLN4 STA   ,U+
 777 0C57    5C              INCB
 778 0C58    81 0D           CMPA  #CR
 779 0C5A    27 0C           BEQ   OUTLN5
 780 0C5C    C1 7C           CMPB  #124
 781 0C5E    25 D6           BCS   OUTLN2
 782 0C60    F7 FC83         STB   SHARED+3
 783 0C63    17 FC86         LBSR  SUBRST
 784 0C66    20 C1           BRA   OUTLN1

 786 0C68    0F 83    OUTLN5 CLR   CNTCHR
 787 0C6A    F7 FC83         STB   SHARED+3
 788 0C6D    17 FC7C         LBSR  SUBRST
 789 0C70    CE FC81         LDU   #SHARED+1
 790 0C73    CC 0302         LDD   #$0302      PUT
 791 0C76    17 FC51         LBSR  SUBHLT
 792 0C79    6F C0           CLR   ,U+
 793 0C7B    ED C1           STD   ,U++
 794 0C7D    86 11           LDA   #SF
 795 0C7F    D6 88           LDB   CRATTR
 796 0C81    ED C4           STD   ,U
 797 0C83    17 FC66         LBSR  SUBRST
 798 0C86    20 A1           BRA   OUTLN1
 799                  *
 800                  *      Read cursor location
 801                  *
 802 0C88    17 FC6F  POS_XY LBSR  GET_XY
 803 0C8B    DC 84           LDD   CSRX
 804 0C8D    ED A810         STD   16,Y        Cursor X-Y pos. (HL)
 805 0C90    39              RTS
 806                  *
 807                  *      Set cursor location
 808                  *
 809 0C91    EC A810  LOCATE LDD   16,Y        Cursor X-Y pos. (HL)
 810 0C94    DD 84           STD   CSRX
 811 0C96    17 FC96         LBSR  CLRCUR      Clear cursor
 812 0C99    17 FC71         LBSR  SET_XY      Set cursor location
 813 0C9C    17 FC2B         LBSR  SUBHLT
 814 0C9F    B6 FC80         LDA   SHARED
 815 0CA2    27 02           BEQ   LOCOK
 816 0CA4    86 01           LDA   #1          If Error Set CarryFlag
 817 0CA6    A7 2A    LOCOK  STA   10,Y
 818 0CA8    86 80           LDA   #$80
 819 0CAA    B7 FC80         STA   SHARED
 820 0CAD    17 FC3C         LBSR  SUBRST
 821 0CB0    16 FC74         LBRA  DSPCUR      Display cursor
 822                  *
 823                  *      Read charactor on screen
 824                  *
 825 0CB3    CE FC82  SCREAD LDU   #SHARED+2
 826 0CB6    86 06           LDA   #$06        GCBLK1
 827 0CB8    17 FC0F         LBSR  SUBHLT
 828 0CBB    A7 C0           STA   ,U+
 829 0CBD    EC A810         LDD   16,Y        Read X-Y pos. (HL)
 830 0CC0    ED C1           STD   ,U++
 831 0CC2    ED C4           STD   ,U
 832 0CC4    17 FC25         LBSR  SUBRST
 833 0CC7    17 FC00         LBSR  SUBHLT
 834 0CCA    B6 FC80         LDA   SHARED
 835 0CCD    26 15           BNE   ERRSCR
 836 0CCF    A7 2A           STA   10,Y        Clear Carry
 837 0CD1    A6 5F           LDA   -1,U
 838 0CD3    81 87           CMPA  #'■
 839 0CD5    26 02           BNE   SCRED1
 840 0CD7    86 7B           LDA   #'(
 841 0CD9    81 F1    SCRED1 CMPA  #PAI
 842 0CDB    26 02           BNE   SCRED2
 843 0CDD    86 7F           LDA   #DEL
 844 0CDF    A7 2B    SCRED2 STA   11,Y        Charactor code (A)
 845 0CE1    16 FBFD         LBRA  RDYREQ

 847 0CE4    86 01    ERRSCR LDA   #$01
 848 0CE6    A7 2A           STA   10,Y
 849 0CE8    16 FBF6         LBRA  RDYREQ

 851                  *
 852                  *      Write charactor to screen
 853                  *
 854 0CEB    CE FC81  SCWRIT LDU   #SHARED+1
 855 0CEE    86 07           LDA   #$07        PCBLK1
 856 0CF0    17 FBD7         LBSR  SUBHLT
 857 0CF3    6F C0           CLR   ,U+
 858 0CF5    A7 C0           STA   ,U+
 859 0CF7    EC A810         LDD   16,Y        Write X-Y pos. (HL)
 860 0CFA    ED C1           STD   ,U++
 861 0CFC    ED C1           STD   ,U++
 862 0CFE    96 88           LDA   CRATTR
 863 0D00    C6 01           LDB   #1
 864 0D02    ED C1           STD   ,U++
 865 0D04    A6 2B           LDA   11,Y        Charactor code (A)
 866 0D06    81 7B           CMPA  #'(
 867 0D08    26 02           BNE   SCWRT1
 868 0D0A    86 87           LDA   #'■
 869 0D0C    81 7F    SCWRT1 CMPA  #DEL
 870 0D0E    26 02           BNE   SCWRT2
 871 0D10    86 F1           LDA   #PAI
 872 0D12    A7 C4    SCWRT2 STA   ,U
 873 0D14    16 FBD5         LBRA  SUBRST
 874                  *
 875                  *      Output 1 charactor to line printer
 876                  *
 877 0D17    A6 2B    OUTLPC LDA   11,Y        Charactor code (A)
 878 0D19    81 0D    OTLPC1 CMPA  #CR
 879 0D1B    26 04           BNE   OTLPC2
 880 0D1D    8D 02           BSR   OTLPC2
 881 0D1F    86 0A           LDA   #LF
 882 0D21    81 7B    OTLPC2 CMPA  #'(
 883 0D23    26 02           BNE   OTLPCB
 884 0D25    86 87           LDA   #'■
 885 0D27    81 7F    OTLPCB CMPA  #DEL
 886 0D29    26 02           BNE   OTLPC3
 887 0D2B    86 F1           LDA   #PAI
 888 0D2D    D6 8A    OTLPC3 LDB   LPRTRY      Retry count
 889 0D2F    34 04           PSHS  B
 890 0D31    8E 3B2F  OTLPCC LDX   #15151      1 sec timer
 891 0D34    F6 FD02  OTLPC4 LDB   MTLPIN
 892 0D37    C5 08           BITB  #%00001000
 893 0D39    26 0F           BNE   OTLPC5
 894 0D3B    C5 02           BITB  #%00000010
 895 0D3D    27 03           BEQ   OTLPCA
 896 0D3F    54              LSRB
 897 0D40    24 11           BCC   OTLPC6
 898 0D42    30 1F    OTLPCA LEAX  -1,X
 899 0D44    26 EE           BNE   OTLPC4
 900 0D46    6A E4           DEC   ,S
 901 0D48    26 E7           BNE   OTLPCC
 902 0D4A    A6 2A    OTLPC5 LDA   10,Y
 903 0D4C    8A 01           ORA   #%00000001
 904 0D4E    A7 2A           STA   10,Y        Set error code (F)
 905 0D50    43              COMA
 906 0D51    35 82           PULS  PC,A

 908 0D53    B7 FD01  OTLPC6 STA   LPDTOT
 909 0D56    96 8B           LDA   MTLPWK
 910 0D58    84 BF           ANDA  #%10111111
 911 0D5A    B7 FD00         STA   MTLPOT
 912 0D5D    8A 40           ORA   #%01000000
 913 0D5F    B7 FD00         STA   MTLPOT
 914 0D62    97 8B           STA   MTLPWK
 915 0D64    A6 2A    OTLPC7 LDA   10,Y
 916 0D66    84 FE           ANDA  #%11111110
 917 0D68    A7 2A           STA   10,Y        (F)
 918 0D6A    4F              CLRA
 919 0D6B    35 82           PULS  PC,A
 920                  *
 921                  *      Output 1 line to line printer
 922                  *
 923 0D6D    EC 2E    OUTLPL LDD   14,Y        Output buffer addr. (DE)
 924 0D6F    1E 89           EXG   A,B
 925 0D71    1F 03           TFR   D,U
 926 0D73    E6 2B           LDB   11,Y        Delimiter (A)
 927 0D75    34 04           PSHS  B
 928 0D77    A6 C0    OTLPL1 LDA   ,U+
 929 0D79    A1 E4           CMPA  ,S
 930 0D7B    27 04           BEQ   OTLPL2
 931 0D7D    8D 9A           BSR   OTLPC1
 932 0D7F    24 F6           BCC   OTLPL1
 933 0D81    35 82    OTLPL2 PULS  PC,A
 934                  *
 935                  *      Input sub-system packet
 936                  *
 937 0D83    EC A810  SUBIN  LDD   16,Y        Packet addr. (HL)
 938 0D86    1E 89           EXG   A,B
 939 0D88    34 06           PSHS  A,B
 940 0D8A    E6 2D           LDB   13,Y        Packet length (B)
 941 0D8C    C1 80           CMPB  #$80
 942 0D8E    23 02           BLS   SUBIN1
 943 0D90    C6 80           LDB   #$80
 944 0D92    8E FC80  SUBIN1 LDX   #SHARED
 945 0D95    35 40           PULS  U
 946 0D97    17 FB30         LBSR  SUBHLT
 947 0D9A    17 FB1C         LBSR  BLKMV1
 948 0D9D    16 FB41         LBRA  RDYREQ
 949                  *
 950                  *      Output sub-system packet
 951                  *
 952 0DA0    EC A810  SUBOUT LDD   16,Y        Packet addr. (HL)
 953 0DA3    1E 89           EXG   A,B
 954 0DA5    1F 01           TFR   D,X
 955 0DA7    E6 2D           LDB   13,Y        Packet length (B)
 956 0DA9    C1 80           CMPB  #$80
 957 0DAB    23 02           BLS   SUBOT1
 958 0DAD    C6 80           LDB   #$80
 959 0DAF    CE FC80  SUBOT1 LDU   #SHARED
 960 0DB2    17 FB15         LBSR  SUBHLT
 961 0DB5    17 FB01         LBSR  BLKMV1
 962 0DB8    16 FB31         LBRA  SUBRST

 964            D0           SETDP $D0
 965                  *
 966                  *      Sub-system test command 2
 967                  *      Read or write video memory
 968                  *
 969 0DBB       00    TST2   FCB   0,0,$3F
 970 0DBE       59           FCC   'YAMAUCHI'
 971 0DC6       93           FCB   $93
 972 0DC7       D38F         FDB   TST21-TST2+_COMAR
 973 0DC9       90           FCB   $90

 975 0DCA       8E    TST21  FCB   $8E         LDX #
 976 0DCB       0002  TST22  RMB   2
 977 0DCD    1F 10           TFR   X,D
 978 0DCF    84 C0           ANDA  #%11000000
 979 0DD1    34 02           PSHS  A
 980 0DD3    DC 1F           LDD   _VOFST
 981 0DD5    84 3F           ANDA  #%00111111
 982 0DD7    C4 E0           ANDB  #%11100000
 983 0DD9    43              COMA
 984 0DDA    53              COMB
 985 0DDB    C3 0001         ADDD  #1
 986 0DDE    30 8B           LEAX  D,X
 987 0DE0    1F 10           TFR   X,D
 988 0DE2    84 3F           ANDA  #%00111111
 989 0DE4    AA E0           ORA   ,S+
 990 0DE6    1F 01           TFR   D,X
 991 0DE8    B5 D409         BITA  VACC
 992 0DEB    20 07    TST23  BRA   TST24
 993 0DED    A6 84           LDA   ,X
 994 0DEF    B7 D383         STA   _COMAR+3
 995 0DF2    20 04           BRA   TST26

 997 0DF4       86    TST24  FCB   $86         LDA #
 998 0DF5       0001  TST25  RMB   1
 999 0DF6    A7 84           STA   ,X
1000 0DF8    B7 D409  TST26  STA   VACC
1001 0DFB    39              RTS

1003            0DFC  TST2EN EQU   *

1005            00           SETDP $00
1006                  *
1007                  *      Video memory read
1008                  *
1009 0DFC    B6 FD05  VMREAD LDA   SBSIN
1010 0DFF    2B FB           BMI   VMREAD
1011 0E01    96 8F           LDA   MPWRK
1012 0E03    8A 07           ORA   #%00000111
1013 0E05    43              COMA
1014 0E06    B7 FD37         STA   MLTPAG
1015 0E09    EC 2E           LDD   14,Y        Video memory addr. (DE)
1016 0E0B    1E 89           EXG   A,B
1017 0E0D    ED 8CBB         STD   TST22,PCR
1018 0E10    86 21           LDA   #$21        BRN
1019 0E12    A7 8CD6         STA   TST23,PCR
1020 0E15    30 8CA3         LEAX  TST2,PCR
1021 0E18    CE FC80         LDU   #SHARED
1022 0E1B    C6 41           LDB   #TST2EN-TST2
1023 0E1D    17 FAAA         LBSR  SUBHLT
1024 0E20    17 FA96         LBSR  BLKMV1
1025 0E23    17 FAC6         LBSR  SUBRST
1026 0E26    17 FAA1         LBSR  SUBHLT
1027 0E29    B6 FC83         LDA   SHARED+3
1028 0E2C    A7 2B           STA   11,Y        Video memory data (A)
1029 0E2E    17 FAB0         LBSR  RDYREQ
1030 0E31    96 8F           LDA   MPWRK
1031 0E33    43              COMA
1032 0E34    B7 FD37         STA   MLTPAG
1033 0E37    39              RTS
1034                  *
1035                  *      Video memory write
1036                  *
1037 0E38    B6 FD05  VMWRIT LDA   SBSIN
1038 0E3B    2B FB           BMI   VMWRIT
1039 0E3D    96 8F           LDA   MPWRK
1040 0E3F    8A 07           ORA   #%00000111
1041 0E41    43              COMA
1042 0E42    B7 FD37         STA   MLTPAG
1043 0E45    EC 2E           LDD   14,Y        Vide momory addr. (DE)
1044 0E47    1E 89           EXG   A,B
1045 0E49    ED 8DFF7E       STD   TST22,PCR
1046 0E4D    86 20           LDA   #$20        BRA
1047 0E4F    A7 8C99         STA   TST23,PCR
```

▶高校生がコンパイラを作成する時代！　年の差を感じます。　矢田　義輝 (42) 京都府

```
1048 0E52   A6 2B              LDA   11,Y       Video memory data (A)
1049 0E54   A7 8C9E            STA   TST25,PCR
1050 0E57   30 8DFF60          LEAX  TST2,PCR
1051 0E5B   CE FC80            LDU   #SHARED
1052 0E5E   C6 41              LDB   #TST2EN-TST2
1053 0E60   17 FA67            LBSR  SUBHLT
1054 0E63   17 FA53            LBSR  BLKMV1
1055 0E66   17 FA83            LBSR  SUBRST
1056 0E69   B6 FD05   VMWRI1   LDA   SBSIN
1057 0E6C   2B FB              BMI   VMWRI1
1058 0E6E   96 8F              LDA   MPWRK
1059 0E70   43                 COMA
1060 0E71   B7 FD37            STA   MLTPAG
1061 0E74   39                 RTS

1063           D0              SETDP $D0
1064                  *
1065                  *        Sub-system test command 3
1066                  *        Read or write sub-system memory
1067                  *
1068 0E75      00     TST3     FCB   0,0,$3F
1069 0E78      59              FCC   'YAMAUCHI'
1070 0E80      93              FCB   $93
1071 0E81      D38F            FDB   TST31-TST3+_COMAR
1072 0E83      90              FCB   $90

1074 0E84      8E     TST31    FCB   $8E        LDX #
1075 0E85      0002   TST32    RMB   2
1076 0E87   B5 D409            BITA  VACC
1077 0E8A   20 07     TST33    BRA   TST34
1078 0E8C   A6 84              LDA   ,X
1079 0E8E   B7 D383            STA   _COMAR+3
1080 0E91   20 04              BRA   TST36

1082 0E93      86     TST34    FCB   $86        LDA #
1083 0E94      0001   TST35    RMB   1
1084 0E95   A7 84              STA   ,X
1085 0E97   B7 D409   TST36    STA   VACC
1086 0E9A   39                 RTS

1088           0E9B   TST3EN   EQU   *

1090           00              SETDP $00
1091                  *
1092                  *        Sub-system memory read
1093                  *
1094 0E9B   EC 2E     SBREAD   LDD   14,Y       Sub-system addr. (DE)
1095 0E9D   1E 89              EXG   A,B
1096 0E9F   ED 8CE3            STD   TST32,PCR
1097 0EA2   86 21              LDA   #$21       BRN
1098 0EA4   A7 8CE3            STA   TST33,PCR
1099 0EA7   30 8CCB            LEAX  TST3,PCR
1100 0EAA   CE FC80            LDU   #SHARED
1101 0EAD   C6 26              LDB   #TST3EN-TST3
1102 0EAF   17 FA18            LBSR  SUBHLT
1103 0EB2   17 FA04            LBSR  BLKMV1
1104 0EB5   17 FA34            LBSR  SUBRST
1105 0EB8   17 FA0F            LBSR  SUBHLT
1106 0EBB   B6 FC83            LDA   SHARED+3
1107 0EBE   A7 2B              STA   11,Y       Sub-system data (A)
1108 0EC0   16 FA1E            LBRA  RDYREQ
1109                  *
1110                  *        Sub-system memory write
1111                  *
1112 0EC3   EC 2E     SBWRIT   LDD   14,Y       Sub-system addr. (DE)
1113 0EC5   1E 89              EXG   A,B
1114 0EC7   ED 8CBB            STD   TST32,PCR
1115 0ECA   86 20              LDA   #$20       BRA
1116 0ECC   A7 8CBB            STA   TST33,PCR
1117 0ECF   A6 2B              LDA   11,Y       Sub-system data (A)
1118 0ED1   A7 8CC0            STA   TST35,PCR
1119 0ED4   30 8C9E            LEAX  TST3,PCR
1120 0ED7   CE FC80            LDU   #SHARED
1121 0EDA   C6 26              LDB   #TST3EN-TST3
1122 0EDC   17 F9EB            LBSR  SUBHLT
1123 0EDF   17 F9D7            LBSR  BLKMV1
1124 0EE2   16 FA07            LBRA  SUBRST

1126           FD              SETDP $FD
1127                  *
1128                  *        5 inch floppy disk read/write
1129                  *
1130 0EE5   86 0A     FDREAD   LDA   #$0A       DREAD
1131 0EE7      8C              FCB   $8C        CMPX #
1132 0EE8   86 09     FDWRIT   LDA   #$09       DWRITE
1133 0EEA   34 09              PSHS  DP,CC
1134 0EEC   C6 FD              LDB   #$FD
1135 0EEE   1F 9B              TFR   B,DP
1136 0EF0   32 78              LEAS  -8,S
1137 0EF2   30 E4              LEAX  ,S
1138 0EF4   A7 84              STA   ,X
1139 0EF6   EC A810            LDD   16,Y       Read/Write buffer addr. (HL)
1140 0EF9   1E 89              EXG   A,B
1141 0EFB   1F 03              TFR   D,U
1142 0EFD   ED 02              STD   2,X
1143 0EFF   A6 2E              LDA   14,Y       Record number (DE)
1144 0F01   84 0F              ANDA  #%00001111
1145 0F03   4C                 INCA
1146 0F04   A7 05              STA   5,X
1147 0F06   A6 2E              LDA   14,Y       Record number (DE)
1148 0F08   5F                 CLRB
1149 0F09   85 10              BITA  #%00010000
1150 0F0B   27 01              BEQ   FDSID0
1151 0F0D   5C                 INCB
1152 0F0E   E7 06     FDSID0   STB   6,X
1153 0F10   EC 2E              LDD   14,Y       Record number (DE)
1154 0F12   1E 89              EXG   A,B
1155 0F14   58                 ASLB             LSLD
1156 0F15   49                 ROLA
1157 0F16   58                 ASLB             LSLD
1158 0F17   49                 ROLA
1159 0F18   58                 ASLB             LSLD
1160 0F19   49                 ROLA
1161 0F1A   84 3F              ANDA  #%00111111
1162 0F1C   A7 04              STA   4,X
1163 0F1E   A6 2C              LDA   12,Y       Unit number (C)
1164 0F20   84 03              ANDA  #%00000011
1165 0F22   A7 07              STA   7,X
1166 0F24   E6 2B              LDB   11,Y       Number of access record (A)
1167 0F26   34 74     FDLOOP   PSHS  U,Y,X,B
1168 0F28   8D 4B              BSR   FDACCS
1169 0F2A   35 74              PULS  U,Y,X,B
1170 0F2C   A7 01              STA   1,X
1171 0F2E   26 23              BNE   FDERR
1172 0F30   5A                 DECB
1173 0F31   27 30              BEQ   FDEXIT
1174 0F33   33 C90100          LEAU  $100,U
1175 0F37   EF 02              STU   2,X
1176 0F39   6C 05              INC   5,X
1177 0F3B   A6 05              LDA   5,X
1178 0F3D   81 10              CMPA  #16
1179 0F3F   23 E5              BLS   FDLOOP
1180 0F41   86 01              LDA   #1
1181 0F43   A7 05              STA   5,X
1182 0F45   6C 06              INC   6,X
1183 0F47   A6 06              LDA   6,X
1184 0F49   81 01              CMPA  #1
1185 0F4B   23 D9              BLS   FDLOOP
1186 0F4D   6F 06              CLR   6,X
1187 0F4F   6C 04              INC   4,X
1188 0F51   20 D3              BRA   FDLOOP

1190 0F53   80 0A     FDERR    SUBA  #10
1191 0F55   27 06              BEQ   FDNRDY
1192 0F57   4A                 DECA
1193 0F58   27 06              BEQ   FDWPTC
1194 0F5A   86 01              LDA   #1
1195 0F5C      8C              FCB   $8C        CMPX #
1196 0F5D   86 02     FDNRDY   LDA   #2
1197 0F5F      8C              FCB   $8C        CMPX #
1198 0F60   86 04     FDWPTC   LDA   #4
1199 0F62      21              FCB   $21        BRN
1200 0F63   4F        FDEXIT   CLRA
1201 0F64   E6 2A              LDB   10,Y       (F)
1202 0F66   A7 2B              STA   11,Y       Error code (A)
1203 0F68   26 03              BNE   FDCRON
1204 0F6A   C4 FE              ANDB  #%11111110
1205 0F6C      8C              FCB   $8C        CMPX #
1206 0F6D   CA 01     FDCRON   ORB   #%00000001
1207 0F6F   E7 2A              STB   10,Y       Error status (F)
1208 0F71   32 68              LEAS  8,S
1209 0F73   35 89              PULS  PC,DP,CC
1210                  *
1211                  *        Floppy disk access sub-routine
1212                  *
1213 0F75   A6 84     FDACCS   LDA   ,X
1214 0F77   81 0A              CMPA  #$0A
1215 0F79   26 03              BNE   FDACCW
1216 0F7B   7E FE08            JMP   DREAD

1218 0F7E   7E FE05   FDACCW   JMP   DWRITE

1220           00              SETDP $00
1221                  *
1222                  *        Call 6809 machine language sub-routine
1223                  *
1224 0F81   EC A810   JSR_HL   LDD   16,Y       Jump addr. (HL)
1225 0F84   1E 89              EXG   A,B
1226 0F86   1F 01              TFR   D,X
1227 0F88   6E 84              JMP   ,X
1228                  *
1229                  *        Boot System
1230                  *
1231 0F8A   17 F93D   NEW_ON   LBSR  SUBHLT
1232 0F8D   30 8D0034          LEAX  CNSINI,PCR
1233 0F91   CE FC80            LDU   #SHARED
1234 0F94   F6 0012            LDB   E_TST4-CNSINI
1235 0F97   A6 80     INILOP   LDA   ,X+
1236 0F99   A7 C0              STA   ,U+
1237 0F9B   5A                 DECB
1238 0F9C   26 F9              BNE   INILOP
1239 0F9E   17 F94B            LBSR  SUBRST
1240 0FA1   30 8D0023          LEAX  TST4,PCR
1241 0FA5   CE FC80            LDU   #SHARED
1242 0FA8   C6 0F              LDB   #E_TST4-TST4
1243 0FAA   A6 80     NEWLOP   LDA   ,X+
1244 0FAC   A7 C0              STA   ,U+
1245 0FAE   5A                 DECB
1246 0FAF   26 F9              BNE   NEWLOP
1247 0FB1   17 F938            LBSR  SUBRST
1248 0FB4   4F                 CLRA
1249 0FB5   B7 FD37            STA   MLTPAG     Screen 7,7
1250 0FB8   8E FD38            LDX   #COLPLT
1251 0FBB   A7 80     PLTINI   STA   ,X+        Color=(i,i) i=0..7
1252 0FBD   4C                 INCA
1253 0FBE   81 07              CMPA  #$07
1254 0FC0   26 F9              BNE   PLTINI
1255 0FC2   7E FE00            JMP   $FE00

1257 0FC5      00     CNSINI   FCB   0,0
1258 0FC7      01              FCB   1

1260 0FC8      00     TST4     FCB   0,0
1261 0FCA      3F              FCB   $3F
1262 0FCB      59              FCC   "YAMAUCHI"
1263 0FD3      93              FCB   $93
1264 0FD4      E000            FDB   $E000
1265 0FD6      90              FCB   $90
1266           0FD7   E_TST4   EQU   *
1267                  *
1268                  *  --- Command jump table ---
1269                  *

1271                  *
1272                  *        System control
1273                  *
1274 0FD7      F922   CMDTBL   FDB   CMDUMY-CMDTBL '     Command error
1275 0FD9      F983            FDB   INIT-CMDTBL '       Init. screen mode
1276 0FDB      F99D            FDB   CNSCTL-CMDTBL '     Console mode control
1277 0FDD      F9B0            FDB   TABSET-CMDTBL '     Console TAB table set
1278 0FDF      F9C8            FDB   SETATR-CMDTBL '     Set console char. attr.
1279 0FE1      F9CF            FDB   MPCTL-CMDTBL '      Multi page control
1280 0FE3      F9DF            FDB   CPCTL-CMDTBL '      Color palette control
1281                  *
1282                  *        Console I/O
1283                  *
1284 0FE5      F9F1            FDB   INCH-CMDTBL '       Input 1 char. with wait
1285 0FE7      FA00            FDB   INCHR-CMDTBL '      Input 1 char. real time
1286 0FE9      FA40            FDB   CHKBRK-CMDTBL '     Check BREAK key status
1287 0FEB      FA53            FDB   OUTCH-CMDTBL '      Output 1 char.
1288 0FED      FADA            FDB   INLN-CMDTBL '       Input 1 line
1289 0FEF      FBAE            FDB   INLNM-CMDTBL '      Input multi line
1290 0FF1      FC3F            FDB   OUTLN-CMDTBL '      Output 1 line
1291 0FF3      FCB1            FDB   POS_XY-CMDTBL '     Read cursor pos.
1292 0FF5      FCBA            FDB   LOCATE-CMDTBL '     Set cursor pos.
1293 0FF7      FCDC            FDB   SCREAD-CMDTBL '     Read char. from screen
1294 0FF9      FD14            FDB   SCWRIT-CMDTBL '     Write char. to screen
1295                  *
1296                  *        Line printer handling
1297                  *
1298 0FFB      FD40            FDB   OUTLPC-CMDTBL '     Output 1 char to "LPT0:
1299 0FFD      FD96            FDB   OUTLPL-CMDTBL '     Output 1 line to "LPT0:
1300                  *
1301                  *        Sub-system I/O
1302                  *
1303 0FFF      FDAC            FDB   SUBIN-CMDTBL '      Input from sub-system
1304 1001      FDC9            FDB   SUBOUT-CMDTBL '     Output to sub-system
1305 1003      FE25            FDB   VMREAD-CMDTBL '     Read data from VRAM
1306 1005      FE61            FDB   VMWRIT-CMDTBL '     Write data to VRAM
1307 1007      FEC4            FDB   SBREAD-CMDTBL '     Read data from sub-system
1308 1009      FEEC            FDB   SBWRIT-CMDTBL '     Write data to sub-system
1309                  *
1310                  *        Floppy disk I/O
1311                  *
1312 100B      FF0E            FDB   FDREAD-CMDTBL '     5" floppy disk read
1313 100D      FF11            FDB   FDWRIT-CMDTBL '     5" floppy disk write
1314                  *
1315                  *        6809 control
1316                  *
1317 100F      F922            FDB   CMDUMY-CMDTBL '     Dummy
1318 1011      FFAA            FDB   JSR_HL-CMDTBL '     Call 6809 sub-routine
1319                  *
1320                  *        System Boot
1321                  *
1322 1013      FFB3            FDB   NEW_ON-CMDTBL '     Boot System
1323                  *
1324                  *  +++ Command handler +++
1325                  *
1326           003E   CMDMAX   EQU   *-CMDTBL

1328                  *        Structure of stack shattle
1329                  *
1330                  *        0,Y 1  2  3  4  5  6  7  8  9
1331                  *         |  |  |  |  |  |  |  |  |  |
1332                  *         F',A',F',A',C',B',E',D',L',H'
1333                  *
1334                  *         10 11 12 13 14 15 16 17 18  19  20  21  22  23
1335                  *         |  |  |  |  |  |  |  |  |   |   |   |   |   |
1336                  *         F ,A ,C ,B ,E ,D ,L ,H ,IX1,IXh,IY1,IYh:PC1,PCh
1337                  *
1338                  *
1339 1015   33 8CBF   CMDHDL   LEAU  CMDTBL,PCR Get command jump table addr.
1340 1018   DC 80              LDD   ZSTKPT     Get interface data
1341 101A   1E 89              EXG   A,B        Addr. adjust
```


▶ウィザードリィがはやっているようですが，私のはX1のマニアタイプなのでウィザードリィができない。しかたないのでファイヤークリスタルをやっているが，いまだに解けない。むずかしい！
青山　豊（17）愛知県

```
1342 101C   1F 02          TFR   D,Y
1343 101E   A6 A4          LDA   ,Y          Extruct F'
1344 1020   D6 82          LDB   FUNCNO      Extruct function code
1345 1022   58             ASLB
1346 1023   C1 3E          CMPB  #CMDMAX     Command error ?
1347 1025   23 04          BLS   CMDNRM      No. normal command
1348 1027   5F             CLRB              Error command set
1349 1028   8A 01          ORA   #%00000001 Set error flag (Carry)
1350 102A      8C          FCB   $8C         CMPX #
1351 102B   84 FE   CMDNRM ANDA  #%11111110 Reset error flag (Carry)
1352 102D   A7 A4          STA   ,Y          Set error condition
1353 102F   EC C5          LDD   B,U
1354 1031   AD CB          JSR   D,U         Call each routine

1356 1033   86 01   CMDENT LDA   #$00000001
1357 1035   B7 FD05        STA   SBSOUT
1358 1038   12             NOP
1359 1039   20 DA          BRA   CMDHDL

1361           103B END09  EQU   *

1363                *
1364                * +++  End of 6809 system module  +++
1365                *

1367 103B      0045        RMB   $1080-*

1369                *
1370                *      Symbol equation
1371                *
1372           0100 DINBUF EQU   $0100       Input buffer ' DINBUF EQU 100H
1373           0300 DBSTK  EQU   $0300       Stack bottom ' DBSTK  EQU 300H
1374           0314 DREGWK EQU   $0300+20    Register save 'DREGWK EQU 300H+20
1375                *
1376                *      Z-80 Debugger version 1.0
1377                *      Coded by Y.Morishita
1378                *
1379 1080      31   DEBUG  FCB   $31         ' DEBUG: LD SP,DBSTK
1380 1081      00          FCB   <DBSTK,>DBSTK
1381 1083      3A          FCB   $3A         ' LD A,(_WIDTH)
1382 1084      5C          FCB   <_WIDTH,>_WIDTH
1383 1086      FE          FCB   $FE,80      ' CP 80
1384 1088      3E          FCB   $3E,80      ' LD A,80
1385 108A      C4          FCB   $C4         ' CALL NZ,WIDCH
1386 108B      AA          FCB   <WIDCH,>WIDCH
1387 108D      11          FCB   $11         ' LD DE,DOPNM
1388 108E      A2          FCB   <DOPNM,>DOPNM
1389 1090      CD          FCB   $CD         ' CALL MSX
1390 1091      13          FCB   <MSX,>MSX
1391 1093      C3          FCB   $C3         ' JP DBMAIN
1392 1094      A2          FCB   <DBMAIN,>DBMAIN
1393                *
1394                *      Debugger work area
1395                *
1396 1096      0004 DORSTK FDB   $0004       Original stack 'DORSTK:DW 0400H
1397 1098      0002 DORADR RMB   2           Return addr. '  DORADR:DS     2
1398 109A      0002 DCRADR RMB   2           Carrent addr. ' DCRADR:DS     2
1399 109C      0002 DEDADR RMB   2           End addr. '     DEDADR:DS     2
1400 109E      00   DBRKF  FCB   0           BP flag '       DBRKL: DB     0
1401 109F      0002 DBRKPT RMB   2           Break point '   DBRKPT:DS     2
1402 10A1      00   DBRKOD FCB   0           (Break point) ' DBRKOD:DB     0
1403                *
1404                *      Maassage
1405                *
1406 10A2      53   DOPNM  FCC   'SWORD-FM7 Z-80 Debugger'
1407 10B9      0D          FCB   CR,EOS

1409 10BB      2D   DBRKM  FCC   '--> Break at $'
1410 10C9      00          FCB   EOS

1412 10CA      57   DSNERM FCC   'What?'
1413 10CF      0D          FCB   CR,EOS

1415 10D1      42   DBRKVM FCC   'Break point is $'
1416 10E1      00          FCB   EOS

1418 10E2      2A   DBRKNM FCC   '*****'
1419 10E6      0D          FCB   CR,EOS

1421 10E8      41   DDMPLM FCC   'Addr > +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B'
1422 1112      20          FCC   ' +C +D +E +F '
1423 111F      3A          FCC   ': Sum  0123456789ABCDEF'
1424 1136      0D          FCB   CR,EOS

1426 1138      20   DDMP1M FCC   ' > '
1427 113B      00          FCB   EOS

1429 113C      3A   DDMP2M FCC   ': '
1430 113E      00          FCB   EOS

1432 113F      20   DDMP3M FCC   '   '
1433 1142      00          FCB   EOS
```

●デバッガのヘルプメニュー

```
1476 15B8      50   DPRTNM FCC   'Printer is enable.'
1477 15CA      0D          FCB   CR,EOS

1479 15CC      50   DPRTFM FCC   'Printer is disenable.'
1480 15E1      0D          FCB   CR,EOS

1482 15E3      20   DREGM  FCC   "     AF    BC    DE    HL    AF'   BC'   DE'   "
1483 160B      48          FCC   "HL'  IX    IY    SP    PC"
1484 1621      0D          FCB   CR
1485 1622      2D          FCC   '-R> '
1486 1626      00          FCB   EOS
1487                *
1488                *      Debugger command jump table
1489                *
1490 1627      F3   DCMDJT FCB   <DBNOP,>DBNOP 'DCMDJT:DW DBNOP    ; A
1491 1629      FB          FCB   <DBBRK,>DBBRK '         DW    DBBRK    ; B
1492 162B      4E          FCB   <DBCAL,>DBCAL '         DW    DBCAL    ; C
1493 162D      A4          FCB   <DBDMP,>DBDMP '         DW    DBDMP    ; D
1494 162F      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; E
1495 1631      3B          FCB   <DBFIL,>DBFIL '         DW    DBFIL    ; F
1496 1633      72          FCB   <DBGO,>DBGO '           DW    DBGO     ; G
1497 1635      A1          FCB   <DBHLP,>DBHLP '         DW    DBHLP    ; H
1498 1637      A7          FCB   <DBINS,>DBINS '         DW    DBINS    ; I
1499 1639      C0          FCB   <DBJMP,>DBJMP '         DW    DBNOP    ; J
1500 163B      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; K
1501 163D      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; L
1502 163F      CB          FCB   <DBMEM,>DBMEM '         DW    DBMEM    ; M
1503 1641      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; N
1504 1643      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; O
1505 1645      1E          FCB   <DBPRT,>DBPRT '         DW    DBPRT    ; P
1506 1647      35          FCB   <DBQIT,>DBQIT '         DW    DBQIT    ; Q
1507 1649      38          FCB   <DBREG,>DBREG '         DW    DBREG    ; R
1508 164B      93          FCB   <DBSEA,>DBSEA '         DW    DBSEA    ; S
1509 164D      DE          FCB   <DBTRF,>DBTRF '         DW    DBTRF    ; T
1510 164F      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; U
1511 1651      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; V
1512 1653      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; W
1513 1655      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; X
1514 1657      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; Y
1515 1659      F3          FCB   <DBNOP,>DBNOP '         DW    DBNOP    ; Z
1516                *
1517                *      Skip space or control char.
1518                *
1519 165B      13   SKPSPL FCB   $13         ' SKPSPL:INC DE
1520 165C      1A   SKPSP  FCB   $1A         ' SKPSP: LD A,(DE)
1521 165D      FE          FCB   $FE,EOS+1   ' CP EOS+1
1522 165F      D8          FCB   $D8         ' RET C
1523 1660      FE          FCB   $FE,SPC+1   ' CP SPC+1
1524 1662      38          FCB   $38         ' JR C,SKPSPL
1525 1663      F7          FCB   SKPSPL-*-1
1526 1664      C9          FCB   $C9         ' RET
1527                *
1528                *      Change char. to upper case
1529                *
1530 1665      E6   UPPER  FCB   $E6,%11011111 'UPPER: AND    11011111B
1531 1667      FE          FCB   $FE,'A      ' CP      "A"
1532 1669      D8          FCB   $D8         ' RET C
1533 166A      FE          FCB   $FE,'Z+1    ' CP      "Z"+1
1534 166C      3F          FCB   $3F         ' CCF
1535 166D      C9          FCB   $C9         ' RET
1536                *
1537                *      Load 2 byte data to HL from (HL)
1538                *
1539 166E      F5   LDHLHL FCB   $F5         ' LDHLHL:PUSH AF
1540 166F      7E          FCB   $7E         ' LD A,(HL)
1541 1670      23          FCB   $23         ' INC HL
1542 1671      66          FCB   $66         ' LD H,(HL)
1543 1672      6F          FCB   $6F         ' LD L,A
1544 1673      F1          FCB   $F1         ' POP AF
1545 1674      C9          FCB   $C9         ' RET
1546                *
1547                *      Debugger break point entry
1548                *
1549 1675      F1   DBBENT FCB   $F1         ' DBBENT:POP AF
1550 1676      ED73        FDB   $ED73       ' LD (DORSTK),SP
1551 1678      96          FCB   <DORSTK,>DORSTK
1552 167A      E3          FCB   $E3         ' EX (SP),HL
1553 167B      22          FCB   $22         ' LD (DORADR),HL
1554 167C      98          FCB   <DORADR,>DORADR
1555 167E      E3          FCB   $E3         ' EX (SP),HL
1556 167F      31          FCB   $31         ' LD SP,DREGWK
1557 1680      14          FCB   <DREGWK,>DREGWK
1558 1682      FDE5        FDB   $FDE5       ' PUSH IY
1559 1684      DDE5        FDB   $DDE5       ' PUSH IX
1560 1686      08          FCB   $08         ' EX AF,AF'
1561 1687      D9          FCB   $D9         ' EXX
1562 1688      E5          FCB   $E5         ' PUSH HL
1563 1689      D5          FCB   $D5         ' PUSH DE
1564 168A      C5          FCB   $C5         ' PUSH BC
1565 168B      F5          FCB   $F5         ' PUSH AF
1566 168C      08          FCB   $08         ' EX AF,AF'
1567 168D      D9          FCB   $D9         ' EXX
1568 168E      E5          FCB   $E5         ' PUSH HL
1569 168F      D5          FCB   $D5         ' PUSH DE
1570 1690      C5          FCB   $C5         ' PUSH BC
1571 1691      F5          FCB   $F5         ' PUSH AF
1572 1692      11          FCB   $11         ' LD DE,DBRKM
1573 1693      BB          FCB   <DBRKM,>DBRKM
1574 1695      CD          FCB   $CD         ' CALL MSX
1575 1696      13          FCB   <MSX,>MSX
1576 1698      2A          FCB   $2A         ' LD HL,(DORADR)
1577 1699      98          FCB   <DORADR,>DORADR
1578 169B      2B          FCB   $2B         ' DEC HL
1579 169C      CD          FCB   $CD         ' CALL PRTHL
1580 169D      E4          FCB   <PRTHL,>PRTHL
1581 169F      CD          FCB   $CD         ' CALL LTNL
1582 16A0      C8          FCB   <LTNL,>LTNL
1583                *
1584                *      Debugger command handler
1585                *
1586 16A2      31   DBMAIN FCB   $31         ' DBMAIN:LD SP,DBSTK
1587 16A3      00          FCB   <DBSTK,>DBSTK
1588 16A5      21          FCB   $21         ' LD HL,DBMAIN
1589 16A6      A2          FCB   <DBMAIN,>DBMAIN
1590 16A8      E5          FCB   $E5         ' PUSH HL
1591 16A9      3E          FCB   $3E,'-      ' LD     A,"-"
1592 16AB      CD          FCB   $CD         ' CALL PRINT
1593 16AC      F5          FCB   <PRINT,>PRINT
1594 16AE      11          FCB   $11         ' LD DE,DINBUF
1595 16AF      00          FCB   <DINBUF,>DINBUF
1596 16B1      CD          FCB   $CD         ' CALL GETL
1597 16B2      37          FCB   <GETL,>GETL
1598 16B4      CD          FCB   $CD         ' CALL SKPSP
1599 16B5      5C          FCB   <SKPSP,>SKPSP
1600 16B7      D8          FCB   $D8         ' RET C
1601 16B8      FE          FCB   $FE,ESC     ' CP ESC
1602 16BA      C8          FCB   $C8         ' RET Z
1603 16BB      FE          FCB   $FE,'-      ' CP      "-"
1604 16BD      20          FCB   $20         ' JR NZ,DBMCNG
1605 16BE      17          FCB   DBMCNG-*-1
1606 16BF      CD          FCB   $CD         ' CALL SKPSPL
1607 16C0      5B          FCB   <SKPSPL,>SKPSPL
1608 16C2      D8          FCB   $D8         ' RET C
1609 16C3      CD          FCB   $CD         ' CALL UPPER
1610 16C4      65          FCB   <UPPER,>UPPER
1611 16C6      38          FCB   $38         ' JR C,DBERR
1612 16C7      2B          FCB   DBERR-*-1
1613 16C8      21          FCB   $21         ' LD HL,DCMDJT
1614 16C9      27          FCB   <DCMDJT,>DCMDJT
1615 16CB      D6          FCB   $D6,'A      ' SUB    "A"
1616 16CD      17          FCB   $17         ' RLA
1617 16CE      4F          FCB   $4F         ' LD C,A
1618 16CF      06          FCB   $06,0       ' LD B,0
1619 16D1      09          FCB   $09         ' ADD HL,BC
1620 16D2      CD          FCB   $CD         ' CALL LDHLHL
1621 16D3      6E          FCB   <LDHLHL,>LDHLHL
1622 16D5      E9          FCB   $E9         ' JP (HL)
1623                *
1624                *      Edit contents of memory
1625                *
1626 16D6      CD   DBMCNG FCB   $CD         ' DBMCNG:CALL HLHEX
1627 16D7      73          FCB   <HLHEX,>HLHEX
1628 16D9      38          FCB   $38         ' JR C,DBERR
1629 16DA      18          FCB   DBERR-*-1
1630 16DB      CD          FCB   $CD         ' CALL SKPSP
1631 16DC      5C          FCB   <SKPSP,>SKPSP
1632 16DE      FE          FCB   $FE,'>      ' CP      '>'
1633 16E0      20          FCB   $20         ' JR NZ,DBERR
1634 16E1      11          FCB   DBERR-*-1
1635 16E2      13          FCB   $13         ' INC DE
1636 16E3      CD   DBMCN1 FCB   $CD         ' DBMCN1:CALL SKPSP
1637 16E4      5C          FCB   <SKPSP,>SKPSP
1638 16E6      D8          FCB   $D8         ' RET C
1639 16E7      FE          FCB   $FE,':      ' CP      ":"
1640 16E9      C8          FCB   $C8         ' RET Z
1641 16EA      CD          FCB   $CD         ' CALL AHEX
1642 16EB      7C          FCB   <AHEX,>AHEX
1643 16ED      38          FCB   $38         ' JR C,DBERR
1644 16EE      04          FCB   DBERR-*-1
1645 16EF      77          FCB   $77         ' LD (HL),A
1646 16F0      23          FCB   $23         ' INC HL
1647 16F1      18          FCB   $18         ' JR DBMCN1
1648 16F2      F0          FCB   DBMCN1-*-1
1649                *
1650                *      Debugger command error trap
1651                *
1652           16F3 DBNOP  EQU   *           ' DBNOP EQU $
1653 16F3      11   DBERR  FCB   $11         ' DBERR: LD DE,DSNERM
1654 16F4      CA          FCB   <DSNERM,>DSNERM
1655 16F6      CD          FCB   $CD         ' CALL MSX
1656 16F7      13          FCB   <MSX,>MSX
1657 16F9      18          FCB   $18         ' JR DBMAIN
1658 16FA      A7          FCB   DBMAIN-*-1
1659                *
1660                *      Break point set/reset/view
1661                *
1662 16FB      2A   DBBRK  FCB   $2A         ' DBBRK: LD HL,(DBRKPT)
1663 16FC      9F          FCB   <DBRKPT,>DBRKPT
1664 16FE      CD          FCB   $CD         ' CALL SKPSPL
1665 16FF      5B          FCB   <SKPSPL,>SKPSPL
1666 1701      30          FCB   $30         ' JR NC,DBBRK3
1667 1702      18          FCB   DBBRK3-*-1
1668 1703      11   DBBRK1 FCB   $11         ' DBBRK1:LD DE,DBRKVM
1669 1704      D1          FCB   <DBRKVM,>DBRKVM
1670 1706      CD          FCB   $CD         ' CALL MSX
1671 1707      13          FCB   <MSX,>MSX
```

▶僕はパソコンと中島みゆきが大好きである。にもかかわらず，巷での僕に対する評価は「明るすぎる○カ」。みなさん，こういう人間もいるんですよ！　こんど，みゆきさんのコンサートに行きまっせ。

小西　徹昌（16）大阪府

```
1672 1709       3A              FCB     $3A           ' LD A,(DBRKF)
1673 170A       9E              FCB     <DBRKF,>DBRKF
1674 170C       B7              FCB     $B7           ' OR A
1675 170D       20              FCB     $20           ' JR NZ,DBBRK2
1676 170E       06              FCB     DBBRK2-*-1
1677 170F       11              FCB     $11           ' LD DE,DBRKNM
1678 1710       E2              FCB     <DBRKNM,>DBRKNM
1679 1712       C3              FCB     $C3           ' JP MSX
1680 1713       13              FCB     <MSX,>MSX

1682 1715       CD      DBBRK2  FCB     $CD           ' DBBRK2:CALL PRTHL
1683 1716       E4              FCB     <PRTHL,>PRTHL
1684 1718       C3              FCB     $C3           ' JP LTNL
1685 1719       C8              FCB     <LTNL,>LTNL

1687 171B       FE      DBBRK3  FCB     $FE,'*        ' DBBRK3 CP     "*"
1688 171D       20              FCB     $20           ' JR NZ,DBBRK4
1689 171E       0F              FCB     DBBRK4-*-1
1690 171F       3A              FCB     $3A           ' LD A,(DBRKF)
1691 1720       9E              FCB     <DBRKF,>DBRKF
1692 1722       B7              FCB     $B7           ' OR A
1693 1723       C8              FCB     $C8           ' RET Z
1694 1724       3A              FCB     $3A           ' LD A,(DBRKOD)
1695 1725       A1              FCB     <DBRKOD,>DBRKOD
1696 1727       77              FCB     $77           ' LD (HL),A
1697 1728       AF              FCB     $AF           ' XOR A
1698 1729       32              FCB     $32           ' LD (DBRKF),A
1699 172A       9E              FCB     <DBRKF,>DBRKF
1700 172C       18              FCB     $18           ' JR DBBRK1
1701 172D       D5              FCB     DBBRK1-*-1

1703 172E       44      DBBRK4  FCB     $44           ' DBBRK4:LD B,H
1704 172F       4D              FCB     $4D           ' LD C,L
1705 1730       CD              FCB     $CD           ' CALL HLHEX
1706 1731       73              FCB     <HLHEX,>HLHEX
1707 1733       38              FCB     $38           ' JR C,DBERR
1708 1734       BE              FCB     DBERR-*-1
1709 1735       3A              FCB     $3A           ' LD A,(DBRKF)
1710 1736       9E              FCB     <DBRKF,>DBRKF
1711 1738       B7              FCB     $B7           ' OR A
1712 1739       28              FCB     $28           ' JR Z,DBBRK5
1713 173A       04              FCB     DBBRK5-*-1
1714 173B       3A              FCB     $3A           ' LD A,(DBRKOD)
1715 173C       A1              FCB     <DBRKOD,>DBRKOD
1716 173E       02              FCB     $02           ' LD (BC),A
1717 173F       7E      DBBRK5  FCB     $7E           ' DBBRK5:LD A,(HL)
1718 1740       32              FCB     $32           ' LD (DBRKOD),A
1719 1741       A1              FCB     <DBRKOD,>DBRKOD
1720 1743       3E              FCB     $3E,$FF       ' LD A,0FFH ; RST 38H
1721 1745       77              FCB     $77           ' LD (HL),A
1722 1746       32              FCB     $32           ' LD (DBRKF),A
1723 1747       9E              FCB     <DBRKF,>DBRKF
1724 1749       22              FCB     $22           ' LD (DBRKPT),HL
1725 174A       9F              FCB     <DBRKPT,>DBRKPT
1726 174C       18              FCB     $18           ' JR DBBRK1
1727 174D       B5              FCB     DBBRK1-*-1
1728                    *
1729                    *       Calc. HEX data
1730                    *
1731 174E       CD      DBCAL   FCB     $CD           ' DBCAL: CALL SKPSPL
1732 174F       5B              FCB     <SKPSPL,>SKPSPL
1733 1751       CD              FCB     $CD           ' CALL HLHEX
1734 1752       73              FCB     <HLHEX,>HLHEX
1735 1754       38              FCB     $38           ' JR C,DBERR
1736 1755       9D              FCB     DBERR-*-1
1737 1756       CD              FCB     $CD           ' CALL SKPSP
1738 1757       5C              FCB     <SKPSP,>SKPSP
1739 1759       38              FCB     $38           ' JR C,DBERR
1740 175A       98              FCB     DBERR-*-1
1741 175B       FE              FCB     $FE,',        ' CP     ',
1742 175D       20              FCB     $20           ' JR NZ,DBERR
1743 175E       94              FCB     DBERR-*-1
1744 175F       44              FCB     $44           ' LD B,H
1745 1760       4D              FCB     $4D           ' LD C,L
1746 1761       CD              FCB     $CD           ' CALL SKPSPL
1747 1762       5B              FCB     <SKPSPL,>SKPSPL
1748 1764       CD              FCB     $CD           ' CALL HLHEX
1749 1765       73              FCB     <HLHEX,>HLHEX
1750 1767       38              FCB     $38           ' JR C,DBERR
1751 1768       8A              FCB     DBERR-*-1
1752 1769       50              FCB     $50           ' LD D,B
1753 176A       59              FCB     $59           ' LD E,C
1754 176B       EB              FCB     $EB           ' EX DE,HL
1755 176C       CD              FCB     $CD           ' CALL PRTHL
1756 176D       E4              FCB     <PRTHL,>PRTHL
1757 176F       E5              FCB     $E5           ' PUSH HL
1758 1770       D5              FCB     $D5           ' PUSH DE
1759 1771       3E              FCB     $3E,'+        ' LD      A,"+"
1760 1773       CD              FCB     $CD           ' CALL PRINT
1761 1774       F5              FCB     <PRINT,>PRINT
1762 1776       EB              FCB     $EB           ' EX DE,HL
1763 1777       CD              FCB     $CD           ' CALL PRTHL
1764 1778       E4              FCB     <PRTHL,>PRTHL
1765 177A       3E              FCB     $3E,'=        ' LD      A,'='
1766 177C       CD              FCB     $CD           ' CALL PRINT
1767 177D       F5              FCB     <PRINT,>PRINT
1768 177F       0019            FDB     $19           ' ADD HL,DE
1769 1781       CD              FCB     $CD           ' CALL PRTHL
1770 1782       E4              FCB     <PRTHL,>PRTHL
1771 1784       D1              FCB     $D1           ' POP DE
1772 1785       E1              FCB     $E1           ' POP HL
1773 1786       CD              FCB     $CD           ' CALL LTNL
1774 1787       C8              FCB     <LTNL,>LTNL
1775 1789       CD              FCB     $CD           ' CALL PRTHL
1776 178A       E4              FCB     <PRTHL,>PRTHL
1777 178C       3E              FCB     $3E,'-        ' LD      A,"-"
1778 178E       CD              FCB     $CD           ' CALL PRINT
1779 178F       F5              FCB     <PRINT,>PRINT
1780 1791       EB              FCB     $EB           ' EX DE,HL
1781 1792       CD              FCB     $CD           ' CALL PRTHL
1782 1793       E4              FCB     <PRTHL,>PRTHL
1783 1795       3E              FCB     $3E,'=        ' LD      A,"="
1784 1797       CD              FCB     $CD           ' CALL PRINT
1785 1798       F5              FCB     <PRINT,>PRINT
1786 179A       EB              FCB     $EB           ' EX DE,HL
1787 179B       B7              FCB     $B7           ' OR A
1788 179C       ED52            FDB     $ED52         ' SBC HL,DE
1789 179E       CD              FCB     $CD           ' CALL PRTHL
1790 179F       E4              FCB     <PRTHL,>PRTHL
1791 17A1       C3              FCB     $C3           ' JP LTNL
1792 17A2       C8              FCB     <LTNL,>LTNL
1793                    *
1794                    *       Dump contents of memory
1795                    *
1796 17A4       2A      DBDMP   FCB     $2A           ' DBDMP: LD HL,(DCRADR)
1797 17A5       9A              FCB     <DCRADR,>DCRADR
1798 17A7       CD              FCB     $CD           ' CALL SKPSPL
1799 17A8       5B              FCB     <SKPSPL,>SKPSPL
1800 17AA       38              FCB     $38           ' JR C,DBDMP1
1801 17AB       09              FCB     DBDMP1-*-1
1802 17AC       CD              FCB     $CD           ' CALL HLHEX
1803 17AD       73              FCB     <HLHEX,>HLHEX
1804 17AF       DA              FCB     $DA           ' JP C,DBERR
1805 17B0       F3              FCB     <DBERR,>DBERR
1806 17B2       22              FCB     $22           ' LD (DCRADR),HL
1807 17B3       9A              FCB     <DCRADR,>DCRADR
1808 17B5       24      DBDMP1  FCB     $24           ' DBDMP2:INC H
1809 17B6       2E              FCB     $2E,0         ' LD L,0
1810 17B8       CD              FCB     $CD           ' CALL SKPSP
1811 17B9       5C              FCB     <SKPSP,>SKPSP
1812 17BB       38              FCB     $38           ' JR C,DBDMP2
1813 17BC       12              FCB     DBDMP2-*-1
1814 17BD       FE              FCB     $FE,',        ' CP      ","
1815 17BF       C2              FCB     $C2           ' JP NZ,DBERR
1816 17C0       F3              FCB     <DBERR,>DBERR
1817 17C2       CD              FCB     $CD           ' CALL SKPSPL
1818 17C3       5B              FCB     <SKPSPL,>SKPSPL
1819 17C5       CD              FCB     $CD           ' CALL HLHEX
1820 17C6       73              FCB     <HLHEX,>HLHEX
1821 17C8       DA              FCB     $DA           ' JP C,DBERR
1822 17C9       F3              FCB     <DBERR,>DBERR
1823 17CB       11              FCB     $11           ' LD DE,0010H
1824 17CC       1000            FDB     $1000
1825 17CE       19              FCB     $19           ' ADD HL,DE
1826 17CF       3E      DBDMP2  FCB     $3E,%11110000 ' DBDMP2: LD A,11110000B
1827 17D1       A5              FCB     $A5           ' AND L
1828 17D2       6F              FCB     $6F           ' LD L,A
1829 17D3       22              FCB     $22           ' LD (DEDADR),HL
1830 17D4       9C              FCB     <DEDADR,>DEDADR
1831 17D6       2A              FCB     $2A           ' LD HL,(DCRADR)
1832 17D7       9A              FCB     <DCRADR,>DCRADR
1833 17D9       3E              FCB     $3E,%11110000 ' LD      A,11110000B
1834 17DB       A5              FCB     $A5           ' AND L
1835 17DC       6F              FCB     $6F           ' LD L,A
1836 17DD       11      DBDMP3  FCB     $11           ' DBDMP3:LD DE,DDMPLM
1837 17DE       E8              FCB     <DDMPLM,>DDMPLM
1838 17E0       CD              FCB     $CD           ' CALL MSX
1839 17E1       13              FCB     <MSX,>MSX
1840 17E3       01      DBDMP4  FCB     $01           ' DBDMP4:LD BC,1000H
1841 17E4       0010            FDB     $0010
1842 17E6       CD              FCB     $CD           ' CALL PRTHL
1843 17E7       E4              FCB     <PRTHL,>PRTHL
1844 17E9       11              FCB     $11           ' LD DE,DDMP1M
1845 17EA       38              FCB     <DDMP1M,>DDMP1M
1846 17EC       CD              FCB     $CD           ' CALL MSX
1847 17ED       13              FCB     <MSX,>MSX
1848 17EF       7E      DBDMP5  FCB     $7E           ' DBDMP5:LD A,(HL)
1849 17F0       81              FCB     $81           ' ADD A,C
1850 17F1       4F              FCB     $4F           ' LD C,A
1851 17F2       7E              FCB     $7E           ' LD A,(HL)
1852 17F3       23              FCB     $23           ' INC HL
1853 17F4       CD              FCB     $CD           ' CALL PRTHX
1854 17F5       E9              FCB     <PRTHX,>PRTHX
1855 17F7       CD              FCB     $CD           ' CALL PRNTS
1856 17F8       DF              FCB     <PRNTS,>PRNTS
1857 17FA       10              FCB     $10           ' DJNZ DBDMP5
1858 17FB       F3              FCB     DBDMP5-*-1
1859 17FC       11              FCB     $11           ' LD DE,DDMP2M
1860 17FD       3C              FCB     <DDMP2M,>DDMP2M
1861 17FF       CD              FCB     $CD           ' CALL MSX
1862 1800       13              FCB     <MSX,>MSX
1863 1802       79              FCB     $79           ' LD A,C
1864 1803       CD              FCB     $CD           ' CALL PRTHX
1865 1804       E9              FCB     <PRTHX,>PRTHX
1866 1806       11              FCB     $11           ' LD DE,DDMP3M
1867 1807       3F              FCB     <DDMP3M,>DDMP3M
1868 1809       CD              FCB     $CD           ' CALL MSX
1869 180A       13              FCB     <MSX,>MSX
1870 180C       11              FCB     $11           ' LD DE,-16
1871 180D       F0              FCB     <$FFF0,>$FFF0
1872 180F       19              FCB     $19           ' ADD HL,DE
1873 1810       06              FCB     $06,16        ' LD B,16
1874 1812       7E      DBDMP6  FCB     $7E           ' DBDMP6:LD A,(HL)
1875 1813       FE              FCB     $FE,' +1      ' CP      " "+1
1876 1815       30              FCB     $30           ' JR NC,DBDMP7
1877 1816       02              FCB     DBDMP7-*-1
1878 1817       3E              FCB     $3E,'         ' LD      A," "
1879 1819       CD      DBDMP7  FCB     $CD           ' DBDMP7:CALL PRINT
1880 181A       F5              FCB     <PRINT,>PRINT
1881 181C       23              FCB     $23           ' INC HL
1882 181D       10              FCB     $10           ' DJNZ DBDMP6
1883 181E       F3              FCB     DBDMP6-*-1
1884 181F       CD              FCB     $CD           ' CALL LTNL
1885 1820       C8              FCB     <LTNL,>LTNL
1886 1822       22              FCB     $22           ' LD (DCRADR),HL
1887 1823       9A              FCB     <DCRADR,>DCRADR
1888 1825       CD              FCB     $CD           ' CALL BRKEY
1889 1826       3A              FCB     <BRKEY,>BRKEY
1890 1828       C8              FCB     $C8           ' RET Z
1891 1829       ED5B            FDB     $ED5B         ' LD DE,(DEDADR)
1892 182B       9C              FCB     <DEDADR,>DEDADR
1893 182D       B7              FCB     $B7           ' OR A
1894 182E       ED52            FDB     $ED52         ' SBC HL,DE
1895 1830       19              FCB     $19           ' ADD HL,DE
1896 1831       C8              FCB     $C8           ' RET Z
1897 1832       7D              FCB     $7D           ' LD A,L
1898 1833       B7              FCB     $B7           ' OR A
1899 1834       20              FCB     $20           ' JR NZ,DBDMP4
1900 1835       AD              FCB     DBDMP4-*-1
1901 1836       CD              FCB     $CD           ' CALL LTNL
1902 1837       C8              FCB     <LTNL,>LTNL
1903 1839       18              FCB     $18           ' JR DBDMP3
1904 183A       A2              FCB     DBDMP3-*-1
1905                    *
1906                    *       Fill memory
1907                    *
1908 183B       CD      DBFIL   FCB     $CD           ' DBFIL: CALL SKPSPL
1909 183C       5B              FCB     <SKPSPL,>SKPSPL
1910 183E       CD              FCB     $CD           ' CALL HLHEX
1911 183F       73              FCB     <HLHEX,>HLHEX
1912 1841       DA              FCB     $DA           ' JP C,DBERR
1913 1842       F3              FCB     <DBERR,>DBERR
1914 1844       CD              FCB     $CD           ' CALL SKPSP
1915 1845       5C              FCB     <SKPSP,>SKPSP
1916 1847       FE              FCB     $FE,',        ' CP      ","
1917 1849       C2              FCB     $C2           ' JP NZ,DBERR
1918 184A       F3              FCB     <DBERR,>DBERR
1919 184C       44              FCB     $44           ' LD B,H
1920 184D       4D              FCB     $4D           ' LD C,L
1921 184E       CD              FCB     $CD           ' CALL SKPSPL
1922 184F       5B              FCB     <SKPSPL,>SKPSPL
1923 1851       CD              FCB     $CD           ' CALL HLHEX
1924 1852       73              FCB     <HLHEX,>HLHEX
1925 1854       DA              FCB     $DA           ' JP C,DBERR
1926 1855       F3              FCB     <DBERR,>DBERR
1927 1857       CD              FCB     $CD           ' CALL SKPSP
1928 1858       5C              FCB     <SKPSP,>SKPSP
1929 185A       FE              FCB     $FE,',        ' CP      ","
1930 185C       C2              FCB     $C2           ' JP NZ,DBERR
1931 185D       F3              FCB     <DBERR,>DBERR
1932 185F       CD              FCB     $CD           ' CALL SKPSPL
1933 1860       5B              FCB     <SKPSPL,>SKPSPL
1934 1862       CD              FCB     $CD           ' CALL AHEX
1935 1863       7C              FCB     <AHEX,>AHEX
1936 1865       DA              FCB     $DA           ' JP C,DBERR
1937 1866       F3              FCB     <DBERR,>DBERR
1938 1868       23              FCB     $23           ' INC HL
1939 1869       02      DBFIL1  FCB     $02           ' DBFIL1:LD (BC),A
1940 186A       03              FCB     $03           ' INC BC
1941 186B       B7              FCB     $B7           ' OR A
1942 186C       ED42            FDB     $ED42         ' SBC HL,BC
1943 186E       09              FCB     $09           ' ADD HL,BC
1944 186F       20              FCB     $20           ' JR NZ,DBFIL1
1945 1870       F8              FCB     DBFIL1-*-1
1946 1871       C9              FCB     $C9           ' RET
1947                    *
1948                    *       Go object program
1949                    *
1950 1872       CD      DBGO    FCB     $CD           ' DBGO: CALL SKPSPL
1951 1873       5B              FCB     <SKPSPL,>SKPSPL
1952 1875       38              FCB     $38           ' JR C,DBGO1
1953 1876       0E              FCB     DBGO1-*-1
1954 1877       CD              FCB     $CD           ' CALL HLHEX
1955 1878       73              FCB     <HLHEX,>HLHEX
1956 187A       DA              FCB     $DA           ' JP C,DBERR
1957 187B       F3              FCB     <DBERR,>DBERR
1958 187D       EB              FCB     $EB           ' EX DE,HL
1959 187E       002A            FDB     $2A           ' LD HL,(DORSTK)
1960 1880       96              FCB     <DORSTK,>DORSTK
1961 1882       73              FCB     $73           ' LD (HL),E
1962 1883       23              FCB     $23           ' INC HL
1963 1884       72              FCB     $72           ' LD (HL),D
1964 1885       31      DBGO1   FCB     $31           ' DBGO1: LD SP,DBSTK
1965 1886       00              FCB     <DBSTK,>DBSTK
```

▶私は，祝一平氏のユニークな人物像，そのひねた性格がにじみ出るような語り口に魅かれるとともに，自らのハード，ソフトに関する知識の僅少さにあきれ，情報処理技術試験をめざしてしまった者である。Oh！MZ は試験にも役立つところもなくはない，と思うのであった。
高橋　伸利（29）福島県

```
1966 1888      F1             FCB   $F1        ' POP AF
1967 1889      C1             FCB   $C1        ' POP BC
1968 188A      D1             FCB   $D1        ' POP DE
1969 188B      E1             FCB   $E1        ' POP HL
1970 188C      08             FCB   $08        ' EX AF,AF'
1971 188D      D9             FCB   $D9        ' EXX
1972 188E      F1             FCB   $F1        ' POP AF
1973 188F      C1             FCB   $C1        ' POP BC
1974 1890      D1             FCB   $D1        ' POP DE
1975 1891      E1             FCB   $E1        ' POP HL
1976 1892      08             FCB   $08        ' EX AF,AF'
1977 1893      D9             FCB   $D9        ' EXX
1978 1894      DDE5           FDB   $DDE5      ' POP IX
1979 1896      FDE5           FDB   $FDE5      ' POP IY
1980 1898      ED7B           FDB   $ED7B      ' LD SP,(DORSTK)
1981 189A      96             FCB   <DORSTK,>DORSTK
1982 189C      E3             FCB   $E3        ' EX (SP),HL
1983 189D      2B             FCB   $2B        ' DEC HL
1984 189E      E3             FCB   $E3        ' EX (SP),HL
1985 189F      FB             FCB   $FB        ' EI
1986 18A0      C9             FCB   $C9        ' RET
1987                   *
1988                   *      Help
1989                   *
1990 18A1      11      DBHLP  FCB   $11        ' DBHLP: LD DE,DHLPM
1991 18A2      43             FCB   <DHLPM,>DHLPM
1992 18A4      C3             FCB   $C3        ' JP MSX
1993 18A5      13             FCB   <MSX,>MSX
1994                   *
1995                   *      Insert string
1996                   *
1997 18A7      CD      DBINS  FCB   $CD        ' DBSIN: CALL SKPSPL
1998 18A8      5B             FCB   <SKPSPL,>SKPSPL
1999 18AA      CD             FCB   $CD        ' CALL HLHEX
2000 18AB      73             FCB   <HLHEX,>HLHEX
2001 18AD      DA             FCB   $DA        ' JP C,DBERR
2002 18AE      F3             FCB   <DBERR,>DBERR
2003 18B0      CD             FCB   $CD        ' CALL SKPSP
2004 18B1      5C             FCB   <SKPSP,>SKPSP
2005 18B3      FE             FCB   $FE,',     ' CP    ","
2006 18B5      C2             FCB   $C2        ' JP NZ,DBERR
2007 18B6      F3             FCB   <DBERR,>DBERR
2008 18B8      13      DBINS1 FCB   $13        ' DBINS1:INC DE
2009 18B9      1A             FCB   $1A        ' LD A,(DE)
2010 18BA      B7             FCB   $B7        ' OR A
2011 18BB      C8             FCB   $C8        ' RET Z
2012 18BC      77             FCB   $77        ' LD (HL),A
2013 18BD      23             FCB   $23        ' INC HL
2014 18BE      18             FCB   $18        ' JR DBINS1
2015 18BF      F8             FCB   DBINS1-*-1
2016                   *
2017                   *      Jump
2018                   *
2019 18C0      CD      DBJMP  FCB   $CD        ' DBJMP: CALL SKPSPL
2020 18C1      5B             FCB   <SKPSPL,>SKPSPL
2021 18C3      CD             FCB   $CD        ' CALL HLHEX
2022 18C4      73             FCB   <HLHEX,>HLHEX
2023 18C6      DA             FCB   $DA        ' JP C,DBERR
2024 18C7      F3             FCB   <DBERR,>DBERR
2025 18C9      FB             FCB   $FB        ' EI
2026 18CA      E9             FCB   $E9        ' JP (HL)
2027                   *
2028                   *      Memory change
2029                   *
2030 18CB      2A      DBMEM  FCB   $2A        ' DBMEM: LD HL,(DCRADR)
2031 18CC      9A             FCB   <DCRADR,>DCRADR
2032 18CE      CD             FCB   $CD        ' CALL SKPSPL
2033 18CF      5B             FCB   <SKPSPL,>SKPSPL
2034 18D1      38             FCB   $38        ' JR C,DBMEM1
2035 18D2      06             FCB   DBMEM1-*-1
2036 18D3      CD             FCB   $CD        ' CALL HLHEX
2037 18D4      73             FCB   <HLHEX,>HLHEX
2038 18D6      DA             FCB   $DA        ' JP C,DBERR
2039 18D7      F3             FCB   <DBERR,>DBERR
2040 18D9      CD      DBMEM1 FCB   $CD        ' DBMEM1:CALL PRTHL
2041 18DA      E4             FCB   <PRTHL,>PRTHL
2042 18DC      3E             FCB   $3E,':     ' LD    A,":"
2043 18DE      CD             FCB   $CD        ' CALL PRINT
2044 18DF      F5             FCB   <PRINT,>PRINT
2045 18E1      7E             FCB   $7E        ' LD A,(HL)
2046 18E2      CD             FCB   $CD        ' CALL PRTHX
2047 18E3      E9             FCB   <PRTHX,>PRTHX
2048 18E5      3E             FCB   $3E,'-     ' LD    A,"-"
2049 18E7      CD             FCB   $CD        ' CALL PRINT
2050 18E8      F5             FCB   <PRINT,>PRINT
2051 18EA      11             FCB   $11        ' LD DE,DINBUF
2052 18EB      00             FCB   <DINBUF,>DINBUF
2053 18ED      CD             FCB   $CD        ' CALL GETL
2054 18EE      37             FCB   <GETL,>GETL
2055 18F0      1A             FCB   $1A        ' LD A,(DE)
2056 18F1      B7             FCB   $B7        ' OR A
2057 18F2      28             FCB   $28        ' JR Z,DBMEM2
2058 18F3      27             FCB   DBMEM2-*-1
2059 18F4      FE             FCB   $FE,ESC    ' CP ESC
2060 18F6      C8             FCB   $C8        ' RET Z
2061 18F7      CD             FCB   $CD        ' CALL SKPSP
2062 18F8      5C             FCB   <SKPSP,>SKPSP
2063 18FA      CD             FCB   $CD        ' CALL HLHEX
2064 18FB      73             FCB   <HLHEX,>HLHEX
2065 18FD      DA             FCB   $DA        ' JP C,DBERR
2066 18FE      F3             FCB   <DBERR,>DBERR
2067 1900      1A             FCB   $1A        ' LD A,(DE)
2068 1901      FE             FCB   $FE,':     ' CP    ":"
2069 1903      C2             FCB   $C2        ' JP NZ,DBERR
2070 1904      F3             FCB   <DBERR,>DBERR
2071 1906      13             FCB   $13        ' INC DE
2072 1907      13             FCB   $13        ' INC DE
2073 1908      13             FCB   $13        ' INC DE
2074 1909      1A             FCB   $1A        ' LD A,(DE)
2075 190A      FE             FCB   $FE,'-     ' CP    "-"
2076 190C      C2             FCB   $C2        ' JP NZ,DBERR
2077 190D      F3             FCB   <DBERR,>DBERR
2078 190F      CD             FCB   $CD        ' CALL SKPSPL
2079 1910      5B             FCB   <SKPSPL,>SKPSPL
2080 1912      38             FCB   $38        ' JR C,DBMEM2
2081 1913      07             FCB   DBMEM2-*-1
2082 1914      CD             FCB   $CD        ' CALL AHEX
2083 1915      7C             FCB   <AHEX,>AHEX
2084 1917      DA             FCB   $DA        ' JP C,DBERR
2085 1918      F3             FCB   <DBERR,>DBERR
2086 191A      77             FCB   $77        ' LD (HL),A
2087 191B      23      DBMEM2 FCB   $23        ' DBMEM2:INC HL
2088 191C      18             FCB   $18        ' JR DBMEM1
2089 191D      BB             FCB   DBMEM1-*-1
2090                   *
2091                   *      Printer switch on/off
2092                   *
2093 191E      3A      DBPRT  FCB   $3A        ' DBPRT: LD A,(_LPSW)
2094 191F      7C             FCB   <_LPSW,>_LPSW
2095 1921      B7             FCB   $B7        ' OR A
2096 1922      20             FCB   $20        ' JR NZ,DBPRT1
2097 1923      01             FCB   DBPRT1-*-1
2098 1924      3E             FCB   $3E        ' DB 3EH ; LD A,nn
2099 1925      AF      DBPRT1 FCB   $AF        ' DBPRT1:XOR A
2100 1926      32             FCB   $32        ' LD (_LPSW),A
2101 1927      7C             FCB   <_LPSW,>_LPSW
2102 1929      11             FCB   $11        ' LD DE,DPRTNM
2103 192A      B8             FCB   <DPRTNM,>DPRTNM
2104 192C      B7             FCB   $B7        ' OR A
2105 192D      20             FCB   $20        ' JR NZ,DBPRT2
2106 192E      03             FCB   DBPRT2-*-1
2107 192F      11             FCB   $11        ' LD DE,DPRTFM
2108 1930      CC             FCB   <DPRTFM,>DPRTFM
2109 1932      C3      DBPRT2 FCB   $C3        ' DBPRT2:JP MSX
2110 1933      13             FCB   <MSX,>MSX
2111                   *
2112                   *      Quit debugger
2113                   *
2114 1935      C3      DBQIT  FCB   $C3        ' DBQIT: JP COLD
2115 1936      29             FCB   <COLD,>COLD
2116                   *
2117                   *      Diplay/Change contents of register
2118                   *
2119 1938      21      DBREG  FCB   $21        ' DBREG: LD HL,DBSTK
2120 1939      00             FCB   <DBSTK,>DBSTK
2121 193B      CD             FCB   $CD        ' CALL SKPSPL
2122 193C      5B             FCB   <SKPSPL,>SKPSPL
2123 193E      30             FCB   $30        ' JR NC,DBREG2
2124 193F      2A             FCB   DBREG2-*-1
2125 1940      11             FCB   $11        ' LD DE,DREGM
2126 1941      E3             FCB   <DREGM,>DREGM
2127 1943      CD             FCB   $CD        ' CALL MSX
2128 1944      13             FCB   <MSX,>MSX
2129 1946      06             FCB   $06,10     ' LD B,10
2130 1948      E5      DBREG1 FCB   $E5        ' DBREG1:PUSH HL
2131 1949      CD             FCB   $CD        ' CALL LDHLHL
2132 194A      6E             FCB   <LDHLHL,>LDHLHL
2133 194C      CD             FCB   $CD        ' CALL PRTHL
2134 194D      E4             FCB   <PRTHL,>PRTHL
2135 194F      E1             FCB   $E1        ' POP HL
2136 1950      CD             FCB   $CD        ' CALL PRNTS
2137 1951      DF             FCB   <PRNTS,>PRNTS
2138 1953      23             FCB   $23        ' INC HL
2139 1954      23             FCB   $23        ' INC HL
2140 1955      10             FCB   $10        ' DJNZ DBREG1
2141 1956      F1             FCB   DBREG1-*-1
2142 1957      2A             FCB   $2A        ' LD HL,(DORSTK)
2143 1958      96             FCB   <DORSTK,>DORSTK
2144 195A      CD             FCB   $CD        ' CALL PRTHL
2145 195B      E4             FCB   <PRTHL,>PRTHL
2146 195D      CD             FCB   $CD        ' CALL PRNTS
2147 195E      DF             FCB   <PRNTS,>PRNTS
2148 1960      CD             FCB   $CD        ' CALL LDHLHL
2149 1961      6E             FCB   <LDHLHL,>LDHLHL
2150 1963      2B             FCB   $2B        ' DEC HL
2151 1964      CD             FCB   $CD        ' CALL PRTHL
2152 1965      E4             FCB   <PRTHL,>PRTHL
2153 1967      C3             FCB   $C3        ' JP LTNL
2154 1968      C8             FCB   <LTNL,>LTNL

2156 196A      FE      DBREG2 FCB   $FE,'>     'DBREG2:CP     ">"
2157 196C      C2             FCB   $C2        ' JP NZ,DBERR
2158 196D      F3             FCB   <DBERR,>DBERR
2159 196F      06             FCB   $06,10     ' LD B,10
2160 1971      CD      DBREG3 FCB   $CD        ' DBREG3:CALL SKPSPL
2161 1972      5B             FCB   <SKPSPL,>SKPSPL
2162 1974      D8             FCB   $D8        ' RET C
2163 1975      E5             FCB   $E5        ' PUSH HL
2164 1976      CD             FCB   $CD        ' CALL HLHEX
2165 1977      73             FCB   <HLHEX,>HLHEX
2166 1979      DA             FCB   $DA        ' JP C,DBERR
2167 197A      F3             FCB   <DBERR,>DBERR
2168 197C      4D             FCB   $4D        ' LD C,L
2169 197D      7C             FCB   $7C        ' LD A,H
2170 197E      E1             FCB   $E1        ' POP HL
2171 197F      71             FCB   $71        ' LD (HL),C
2172 1980      23             FCB   $23        ' INC HL
2173 1981      77             FCB   $77        ' LD (HL),A
2174 1982      23             FCB   $23        ' INC HL
2175 1983      10             FCB   $10        ' DJNZ DBREG3
2176 1984      EC             FCB   DBREG3-*-1
2177 1985      CD             FCB   $CD        ' CALL SKPSPL
2178 1986      5B             FCB   <SKPSPL,>SKPSPL
2179 1988      D8             FCB   $D8        ' RET C
2180 1989      CD             FCB   $CD        ' CALL HLHEX
2181 198A      73             FCB   <HLHEX,>HLHEX
2182 198C      DA             FCB   $DA        ' JP C,DBERR
2183 198D      F3             FCB   <DBERR,>DBERR
2184 198F      22             FCB   $22        ' LD (DORSTK),HL
2185 1990      96             FCB   <DORSTK,>DORSTK
2186 1992      C9             FCB   $C9        ' RET
2187                   *
2188                   *      Search
2189                   *
2190 1993      CD      DBSEA  FCB   $CD        ' DBSEA: CALL SKPSPL
2191 1994      5B             FCB   <SKPSPL,>SKPSPL
2192 1996      CD             FCB   $CD        ' CALL HLHEX
2193 1997      73             FCB   <HLHEX,>HLHEX
2194 1999      DA             FCB   $DA        ' JP C,DBERR
2195 199A      F3             FCB   <DBERR,>DBERR
2196 199C      44             FCB   $44        ' LD B,H
2197 199D      4D             FCB   $4D        ' LD C,L
2198 199E      CD             FCB   $CD        ' CALL SKPSP
2199 199F      5C             FCB   <SKPSP,>SKPSP
2200 19A1      FE             FCB   $FE,',     ' CP    ","
2201 19A3      C2             FCB   $C2        ' JP NZ,DBERR
2202 19A4      F3             FCB   <DBERR,>DBERR
2203 19A6      CD             FCB   $CD        ' CALL SKPSPL
2204 19A7      5B             FCB   <SKPSPL,>SKPSPL
2205 19A9      CD             FCB   $CD        ' CALL HLHEX
2206 19AA      73             FCB   <HLHEX,>HLHEX
2207 19AC      DA             FCB   $DA        ' JP C,DBERR
2208 19AD      F3             FCB   <DBERR,>DBERR
2209 19AF      CD             FCB   $CD        ' CALL SKPSP
2210 19B0      5C             FCB   <SKPSP,>SKPSP
2211 19B2      FE             FCB   $FE,',     ' CP    ","
2212 19B4      C2             FCB   $C2        ' JP NZ,DBERR
2213 19B5      F3             FCB   <DBERR,>DBERR
2214 19B7      CD             FCB   $CD        ' CALL SKPSPL
2215 19B8      5B             FCB   <SKPSPL,>SKPSPL
2216 19BA      CD             FCB   $CD        ' CALL AHEX
2217 19BB      7C             FCB   <AHEX,>AHEX
2218 19BD      DA             FCB   $DA        ' JP C,DBERR
2219 19BE      F3             FCB   <DBERR,>DBERR
2220 19C0      50             FCB   $50        ' LD D,B
2221 19C1      59             FCB   $59        ' LD E,C
2222 19C2      EB             FCB   $EB        ' EX DE,HL
2223 19C3      13             FCB   $13        ' INC DE
2224 19C4      BE      DBSEA1 FCB   $BE        ' DBSEA1:CP (HL)
2225 19C5      20             FCB   $20        ' JR NZ,DBSEA2
2226 19C6      08             FCB   DBSEA2-*-1
2227 19C7      08             FCB   $08        ' EX AF,AF'
2228 19C8      CD             FCB   $CD        ' CALL PRTHL
2229 19C9      E4             FCB   <PRTHL,>PRTHL
2230 19CB      CD             FCB   $CD        ' CALL PRNTS
2231 19CC      DF             FCB   <PRNTS,>PRNTS
2232 19CE      08             FCB   $08        ' EX AF,AF'
2233 19CF      23      DBSEA2 FCB   $23        ' DBSEA2:INC HL
2234 19D0      CD             FCB   $CD        ' CALL BRKEY
2235 19D1      3A             FCB   <BRKEY,>BRKEY
2236 19D3      28             FCB   $28        ' JR Z,DBSEA3
2237 19D4      06             FCB   DBSEA3-*-1
2238 19D5      B7             FCB   $B7        ' OR A
2239 19D6      ED52           FDB   $ED52      ' SBC HL,DE
2240 19D8      19             FCB   $19        ' ADD HL,DE
2241 19D9      20             FCB   $20        ' JR NZ,DBSEA1
2242 19DA      E9             FCB   DBSEA1-*-1
2243 19DB      C3      DBSEA3 FCB   $C3        ' DBSEA3:JP LTNL
2244 19DC      C8             FCB   <LTNL,>LTNL
2245                   *
2246                   *      Transfer
2247                   *
2248 19DE      CD      DBTRF  FCB   $CD        ' DBTRF: CALL SKPSPL
2249 19DF      5B             FCB   <SKPSPL,>SKPSPL
2250 19E1      CD             FCB   $CD        ' CALL HLHEX
2251 19E2      73             FCB   <HLHEX,>HLHEX
2252 19E4      DA             FCB   $DA        ' JP C,DBERR
2253 19E5      F3             FCB   <DBERR,>DBERR
2254 19E7      CD             FCB   $CD        ' CALL SKPSP
2255 19E8      5C             FCB   <SKPSP,>SKPSP
2256 19EA      FE             FCB   $FE,',     ' CP    ","
2257 19EC      C2             FCB   $C2        ' JP NZ,DBERR
2258 19ED      F3             FCB   <DBERR,>DBERR
2259 19EF      22             FCB   $22        ' LD (DCRADR),HL
```

▶１年６カ月ぶりにOh！MZを買いました。それは，今年の３月号にS-OS“SWORD”の再掲載があったのを知ったからです。さっそく３月号を本屋さんに注文しました。来月号からはきっと肩身の広い思いができるでしょう。やはりOh！MZは他誌とは違うなにかがあるっ！
藤田　康一（16）静岡県

```
2260 19F0         9A            FCB   <DCRADR,>DCRADR
2261 19F2         CD            FCB   $CD        ' CALL SKPSPL
2262 19F3         5B            FCB   <SKPSPL,>SKPSPL
2263 19F5         CD            FCB   $CD        ' CALL HLHEX
2264 19F6         73            FCB   <HLHEX,>HLHEX
2265 19F8         DA            FCB   $DA        ' JP C,DBERR
2266 19F9         F3            FCB   <DBERR,>DBERR
2267 19FB         CD            FCB   $CD        ' CALL SKPSP
2268 19FC         5C            FCB   <SKPSP,>SKPSP
2269 19FE         FE            FCB   $FE,'.     ' CP    ","
2270 1A00         C2            FCB   $C2        ' JP C,DBERR
2271 1A01         F3            FCB   <DBERR,>DBERR
2272 1A03         22            FCB   $22        ' LD (DEDADR),HL
2273 1A04         9C            FCB   <DEDADR,>DEDADR
2274 1A06         CD            FCB   $CD        ' CALL SKPSPL
2275 1A07         5B            FCB   <SKPSPL,>SKPSPL
2276 1A09         CD            FCB   $CD        ' CALL HLHEX
2277 1A0A         73            FCB   <HLHEX,>HLHEX
2278 1A0C         DA            FCB   $DA        ' JP C,DBERR
2279 1A0D         F3            FCB   <DBERR,>DBERR
2280 1A0F         ED5B          FDB   $ED5B      ' LD DE,(DEDADR)
2281 1A11         9C            FCB   <DEDADR,>DEDADR
2282 1A13         13            FCB   $13        ' INC DE
2283 1A14         B7            FCB   $B7        ' OR A
2284 1A15         ED52          FDB   $ED52      ' SBC HL,DE
2285 1A17         19            FCB   $19        ' ADD HL,DE
2286 1A18         30            FCB   $30        ' JR NC,DBTRF1
2287 1A19         23            FCB   DBTRF1-*-1
2288 1A1A         ED5B          FDB   $ED5B      ' LD DE,(DCRADR)
2289 1A1C         9A            FCB   <DCRADR,>DCRADR
2290 1A1E         B7            FCB   $B7        ' OR A
2291 1A1F         ED52          FDB   $ED52      ' SBC HL,DE
2292 1A21         19            FCB   $19        ' ADD HL,DE
2293 1A22         C8            FCB   $C8        ' RET Z
2294 1A23         38            FCB   $38        ' JR C,DBTRF1
2295 1A24         18            FCB   DBTRF1-*-1
2296 1A25         EB            FCB   $EB        ' EX DE,HL
2297 1A26         2A            FCB   $2A        ' LD HL,(DEDADR)
2298 1A27         9C            FCB   <DEDADR,>DEDADR
2299 1A29         ED4B          FDB   $ED4B      ' LD BC,(DCRADR)
2300 1A2B         9A            FCB   <DCRADR,>DCRADR
2301 1A2D         B7            FCB   $B7        ' OR A
2302 1A2E         ED42          FDB   $ED42      ' SBC HL,BC
2303 1A30         44            FCB   $44        ' LD B,H
2304 1A31         4D            FCB   $4D        ' LD C,L
2305 1A32         62            FCB   $62        ' LD H,D
2306 1A33         6B            FCB   $6B        ' LD L,E
2307 1A34         09            FCB   $09        ' ADD HL,BC
2308 1A35         EB            FCB   $EB        ' EX DE,HL
2309 1A36         2A            FCB   $2A        ' LD HL,(DEDADR)
2310 1A37         9C            FCB   <DEDADR,>DEDADR
2311 1A39         03            FCB   $03        ' INC BC
2312 1A3A         EDB8          FDB   $EDB8      ' LDDR
2313 1A3C         C9            FCB   $C9        ' RET

2315 1A3D         54     DBTRF1 FCB   $54        ' DBTRF1;LD D,H
2316 1A3E         5D            FCB   $5D        ' LD E,L
2317 1A3F         2A            FCB   $2A        ' LD HL,(DEDADR)
2318 1A40         9C            FCB   <DEDADR,>DEDADR
2319 1A42         ED4B          FDB   $ED4B      ' LD BC,(DCRADR)
2320 1A44         9A            FCB   <DCRADR,>DCRADR
2321 1A46         B7            FCB   $B7        ' OR A
2322 1A47         ED42          FDB   $ED42      ' SBC HL,BC
2323 1A49         44            FCB   $44        ' LD B,H
2324 1A4A         4D            FCB   $4D        ' LD C,L
2325 1A4B         2A            FCB   $2A        ' LD HL,(DCRADR)
2326 1A4C         9A            FCB   <DCRADR,>DCRADR
2327 1A4E         03            FCB   $03        ' INC BC
2328 1A4F         EDB0          FDB   $EDB0      ' LDIR
2329 1A51         C9            FCB   $C9        ' RET

2331              1A52   ENDDEB EQU   *

2333                     *
2334                     *      End of Z-80 debugger module
2335                     *


2338                     *
2339                     * +++  Z-80 system module  +++
2340                     *
2341                     *      Designed by Y.Morishita
2342                     *      Coded by Y.Morishita
2343                     *
2344 1A52         00AE          RMB   $1B00-*
2345                     *
2346                     *      Restart hook
2347                     *
2348 1B00         C3     ZHK00H FCB   $C3        ' ZHK00H:JP COLD ; Cold star
2349 1B01         29            FCB   <COLD,>COLD
2350 1B03         C3     ZHK08H FCB   $C3        ' ZHK08H:JP CMDGO ; CMD handl
2351 1B04         0C            FCB   <CMDGO,>CMDGO
2352 1B06         C3     ZHK10H FCB   $C3        ' ZHK10H:JP RSTDMY ; Dummy
2353 1B07         1B            FCB   <RSTDMY,>RSTDMY
2354 1B09         C3     ZHK18H FCB   $C3        ' ZHK18H:JP RSTDMY ; Dummy
2355 1B0A         1B            FCB   <RSTDMY,>RSTDMY
2356 1B0C         C3     ZHK20H FCB   $C3        ' ZHK20H:JP RSTDMY ; Dummy
2357 1B0D         1B            FCB   <RSTDMY,>RSTDMY
2358 1B0F         C3     ZHK28H FCB   $C3        ' ZHK28H:JP RSTDMY ; Dummy
2359 1B10         1B            FCB   <RSTDMY,>RSTDMY
2360 1B12         C3     ZHK30H FCB   $C3        ' ZHK30H:JP RSTDMY ; Dummy
2361 1B13         1B            FCB   <RSTDMY,>RSTDMY
2362 1B15         C3     ZHK38H FCB   $C3        ' ZHK38H:JP RST38H ; Break
2363 1B16         1C            FCB   <RST38H,>RST38H
2364 1B18         C3     ZHK66H FCB   $C3        ' ZHK66H:JP RSTDMY ; Dummy
2365 1B19         1B            FCB   <RSTDMY,>RSTDMY
2366                     *
2367                     *      Restart dummy GAP
2368                     *
2369 1B1B         C9     RSTDMY FCB   $C9        ' RSTDMY:RET
2370                     *
2371                     *      RST 38H GAP
2372                     *
2373 1B1C         F5     RST38H FCB   $F5        ' RST38H:PUSH AF
2374 1B1D         ED57          FDB   $ED57      ' LD A,I
2375 1B1F         EA            FCB   $EA        ' JP PE,DBBENT
2376 1B20         75            FCB   <DBBENT,>DBBENT
2377 1B22         CF            FCB   $CF,0      ' SWORD S$DMYCMD
2378 1B24         F1            FCB   $F1        ' POP AF
2379 1B25         FB            FCB   $FB        ' EI
2380 1B26         C9            FCB   $C9        ' RET
2381                     *
2382                     *      NMI dummy GAP
2383                     *
2384 1B27         ED45   NMIDMY FDB   $ED45      ' NMIDMY:RETN
2385                     *
2386                     * ---- Cold start ----
2387                     *
2388 1B29         F3     COLD   FCB   $F3        ' COLD: DI
2389 1B2A         ED56          FDB   $ED56      ' IM 1
2390 1B2C         31            FCB   $31        ' LD SP,Z80STK
2391 1B2D         00            FCB   <Z80STK,>Z80STK
2392 1B2F         3E            FCB   $3E,80     ' LD A,80
2393 1B31         CD            FCB   $CD        ' CALL WIDCH
2394 1B32         AA            FCB   <WIDCH,>WIDCH
2395 1B34         3E            FCB   $3E,%00111111 ' LD     A,00111111B
2396 1B36         CF            FCB   $CF,2      ' SWORD S$CNSCTL
2397 1B38         21            FCB   $21        ' LD HL,TABMAP
2398 1B39         8C            FCB   <TABMAP,>TABMAP
2399 1B3B         CF            FCB   $CF,3      ' SWORD S$TABSET
2400 1B3D         3E            FCB   $3E,7      ' LD A,7
2401 1B3F         CF            FCB   $CF,4      ' SWORD S$SETATR
2402 1B41         3E            FCB   $3E,$44    ' LD A,44H
2403 1B43         CF            FCB   $CF,5      ' SWORD S$MPCTL
2404 1B45         21            FCB   $21        ' LD HL,CPMAP
2405 1B46         96            FCB   <CPMAP,>CPMAP
2406 1B48         06            FCB   $06,8      ' LD B,8
2407 1B4A         7E     COLD1  FCB   $7E        ' COLD1: LD A,(HL)
2408 1B4B         23            FCB   $23        ' INC HL
2409 1B4C         CF            FCB   $CF,6      ' SWORD S$CPCTL
2410 1B4E         10            FCB   $10        ' DJNZ COLD1
2411 1B4F         FA            FCB   COLD1-*-1
2412 1B50         21            FCB   $21        ' LD HL,8000H
2413 1B51         0080          FDB   $0080
2414 1B53         22            FCB   $22        ' LD (_WKSIZ),HL
2415 1B54         68            FCB   <_WKSIZ,>_WKSIZ
2416 1B56         AF            FCB   $AF        ' XOR A
2417 1B57         32            FCB   $32        ' LD (_LPSW),A
2418 1B58         7C            FCB   <_LPSW,>_LPSW
2419 1B5A         32            FCB   $32        ' LD (_DVSW),A
2420 1B5B         7D            FCB   <_DVSW,>_DVSW
2421 1B5D         21            FCB   $21        ' LD HL,22*$100
2422 1B5E         00     COLD2  FCB   0          '
2423 1B5F         16     COLD21 FCB   22
2424 1B60         CF            FCB   $CF,15     ' SWORD S$LOCATE
2425 1B62         CD            FCB   $CD        ' CALL MPRNT
2426 1B63         FF            FCB   <MPRNT,>MPRNT
2427 1B65         0C            FCB   EA
2428 1B66         3C            FCC   '<<<<< S-OS SWORD >>>>>'
2429 1B7C         0D            FCB   CR,EOS
2430 1B7E         3E            FCB   $3E,1      ' LD A,1
2431 1B80         32            FCB   $32        ' LD (CIPARM+6),A
2432 1B81         A8            FCB   <CIPAR6,>CIPAR6
2433 1B83         AF            FCB   $AF        ' XOR A
2434 1B84         32            FCB   $32        ' LD (COLD2+1),A
2435 1B85         5F            FCB   <COLD21,>COLD21
2436 1B87         FB            FCB   $FB        ' EI
2437 1B88         2A            FCB   $2A        ' LD HL,(_USR)
2438 1B89         7E            FCB   <_USR,>_USR
2439 1B8B         E9            FCB   $E9        ' JP (HL)
2440                     *
2441                     *      Console TAB bit map
2442                     *
2443 1B8C         8080   TABMAP FDB   %1000000010000000
2444 1B8E         8080          FDB   %1000000010000000
2445 1B90         8080          FDB   %1000000010000000
2446 1B92         8080          FDB   %1000000010000000
2447 1B94         8080          FDB   %1000000010000000
2448                     *
2449                     *      Color palette map
2450                     *
2451 1B96         00     CPMAP  FCB   $00        ' CPMAP: DB 00H
2452 1B97         10            FCB   $10        ' DB 10H
2453 1B98         20            FCB   $20        ' DB 20H
2454 1B99         30            FCB   $30        ' DB 30H
2455 1B9A         47            FCB   $47        ' DB 47H
2456 1B9B         57            FCB   $57        ' DB 57H
2457 1B9C         67            FCB   $67        ' DB 67H
2458 1B9D         77            FCB   $77        ' DB 77H
2459                     *
2460                     *      Version
2461                     *
2462 1B9E         21     VER    FCB   $21        ' VER: LD HL,4020H
2463 1B9F         2040          FDB   $2040      ' ; FM-7 Ver.2
2464 1BA1         C9            FCB   $C9        ' RET
2465                     *
2466                     *      Z-80 Command handler
2467                     *

2469                     *
2470                     *      Console initialize parameter
2471                     *
2472 1BA2         00     CIPARM FCB   0          ' CIPARM:DB 0 ; Back color
2473 1BA3         50     CIPAR1 FCB   80         ' DB 80 ; Columbs
2474 1BA4         19            FCB   25         ' DB 25 ; Lines
2475 1BA5         00            FCB   0          ' DB 0 ; Scroll start
2476 1BA6         19            FCB   25         ' DB 25 ; Scroll lines
2477 1BA7         00            FCB   0          ' DB 0 ; Disp. PF key
2478 1BA8         00     CIPAR6 FCB   0          ' DB 0 ; Erase flag
2479 1BA9         01            FCB   1          ' DB 1 ; Green mode
2480                     *
2481                     *      Console width change
2482                     *
2483                     *      IN     A:Console width (80 or 40)
2484                     *      OUT    /
2485                     *      BRK    AF,HL
2486                     *
2487 1BAA         21     WIDCH  FCB   $21        ' WIDCH: LD HL,CIPARM+1
2488 1BAB         A3            FCB   <CIPAR1,>CIPAR1
2489 1BAD         FE            FCB   $FE,40+1   ' CP 40+1
2490 1BAF         3E            FCB   $3E,40     ' LD A,40
2491 1BB1         38            FCB   $38        ' JR C,WIDCH4
2492 1BB2         01            FCB   WIDCH4-*-1
2493 1BB3         17            FCB   $17        ' RLA
2494 1BB4         32     WIDCH4 FCB   $32        ' LD (_WIDTH),A
2495 1BB5         5C            FCB   <_WIDTH,>_WIDTH
2496 1BB7         77            FCB   $77        ' LD (HL),A
2497 1BB8         2B            FCB   $2B        ' DEC HL
2498 1BB9         CF            FCB   $CF,1      ' SWORD S$INIT
2499 1BBB         B7            FCB   $B7        ' OR A
2500 1BBC         C9            FCB   $C9        ' RET
2501                     *
2502                     *      When cursor X location is not zero,output CR
2503                     *
2504                     *      IN     /
2505                     *      OUT    /
2506                     *      BRK    /
2507                     *
2508 1BBD         F5     NL     FCB   $F5        ' NL: PUSH AF
2509 1BBE         E5            FCB   $E5        ' PUSH HL
2510 1BBF         CF            FCB   $CF,14     ' SWORD S$POS_XY
2511 1BC1         7D            FCB   $7D        ' LD A,L
2512 1BC2         E1            FCB   $E1        ' POP HL
2513 1BC3         B7            FCB   $B7        ' OR A
2514 1BC4         20            FCB   $20        ' JR NZ,LTNL1
2515 1BC5         03            FCB   LTNL1-*-1
2516 1BC6         F1            FCB   $F1        ' POP AF
2517 1BC7         C9            FCB   $C9        ' RET
2518                     *
2519                     *      Output carriage return
2520                     *
2521                     *      IN     /
2522                     *      OUT    /
2523                     *      BRK    /
2524                     *
2525 1BC8         F5     LTNL   FCB   $F5        ' LTNL: PUSH AF
2526 1BC9         3E     LTNL1  FCB   $3E,CR     ' LTNL1: LD A,CR
2527 1BCB         18            FCB   $18        ' JR PRINT1
2528 1BCC         29            FCB   PRINT1-*-1
2529                     *
2530                     *      Sing BELL
2531                     *
2532                     *      IN     /
2533                     *      OUT    /
2534                     *      BRK    /
2535                     *
2536 1BCD         F5     BELL   FCB   $F5        ' BELL: PUSH AF
2537 1BCE         3E            FCB   $3E,BEL    ' LD A,BEL
2538 1BD0         18            FCB   $18        ' JR PRINT1
2539 1BD1         24            FCB   PRINT1-*-1
2540                     *
2541                     *      Tabulation
2542                     *
2543                     *      IN     /
2544                     *      OUT    /
2545                     *      BRK    AF
2546                     *
2547 1BD2         3A     TAB    FCB   $3A        ' TAB: LD A,(CNTCHR)
2548 1BD3         83            FCB   <CNTCHR,>CNTCHR
2549 1BD5         90            FCB   $90        ' SUB B
2550 1BD6         3F            FCB   $3F        ' CCF
2551 1BD7         D8            FCB   $D8        ' RET C
2552 1BD8         CD     TAB1   FCB   $CD        ' TAB1: CALL PRNTS
2553 1BD9         DF            FCB   <PRNTS,>PRNTS
```

▶このとおり頭を下げます,石上達也様。PC-8001mkII用“SWORD”を発表してください。X68000の発表からOh！MZに移ってきました。「ひかり」と「やえもん」の差というのがとても気に入ったんですが,「やえもん」を知っているのは読者の何%くらいいるのでしょうね。　佐川　正人（18）北海道

```
2554 1BDB        3C              FCB   $3C          ' INC A
2555 1BDC        2Ø              FCB   $2Ø          ' JR NZ,TAB1
2556 1BDD        FA              FCB   TAB1-*-1
2557 1BDE        C9              FCB   $C9          ' RET
2558                      *
2559                      *      Display space
2560                      *
2561                      *      IN     /
2562                      *      OUT    /
2563                      *      BRK    /
2564                      *
2565 1BDF        F5       PRNTS  FCB   $F5          ' PRNTS: PUSH AF
2566 1BEØ        3E              FCB   $3E,SPC      ' LD A,SPC
2567 1BE2        18              FCB   $18          ' JR PRINT1
2568 1BE3        12              FCB   PRINT1-*-1
2569                      *
2570                      *      Display hexa-decimal 4 charactor in HL
2571                      *
2572                      *      IN     HL:Binary data
2573                      *      OUT    /
2574                      *      BRK    AF
2575                      *
2576 1BE4        7C       PRTHL  FCB   $7C          ' PRTHL: LD A,H
2577 1BE5        CD              FCB   $CD          ' CALL PRTHX
2578 1BE6        E9              FCB   <PRTHX,>PRTHX
2579 1BE8        7D              FCB   $7D          ' LD A,L
2580                      *
2581                      *      Display hexa-decimal 2 charactor in Acc.
2582                      *
2583                      *      IN     A:Binary data
2584                      *      OUT    /
2585                      *      BRK    AF
2586                      *
2587 1BE9        F5       PRTHX  FCB   $F5          ' PRTHX: PUSH AF
2588 1BEA        ØF              FCB   $ØF          ' RRCA
2589 1BEB        ØF              FCB   $ØF          ' RRCA
2590 1BEC        ØF              FCB   $ØF          ' RRCA
2591 1BED        ØF              FCB   $ØF          ' RRCA
2592 1BEE        CD              FCB   $CD          ' CALL PRTHX1
2593 1BEF        F2              FCB   <PRTHX1,>PRTHX1
2594 1BF1        F1              FCB   $F1          ' POP AF
2595 1BF2        CD       PRTHX1 FCB   $CD          ' PRTHX1:CALL ASC
2596 1BF3        6B              FCB   <ASC,>ASC
2597                      *
2598                      *      Output 1 charactor in Acc.
2599                      *
2600                      *      IN     A:Data
2601                      *      OUT    /
2602                      *      BRK    /
2603                      *
2604 1BF5        F5       PRINT  FCB   $F5          ' PRINT: PUSH AF
2605 1BF6        CF       PRINT1 FCB   $CF,1Ø       ' PRINT1:SWORD S$OUTCH
2606 1BF8        CD              FCB   $CD          ' CALL CHKLPS
2607 1BF9        1E              FCB   <CHKLPS,>CHKLPS
2608 1BFB        CF              FCB   $CF,18       ' SWORD O$OUTLPC
2609 1BFD        F1              FCB   $F1          ' POP AF
2610 1BFE        C9              FCB   $C9          ' RET
2611                      *
2612                      *      Output string inline parameter operation
2613                      *
2614                      *      IN     (SP):String top addr.
2615                      *      OUT    (SP):inc. return addr.
2616                      *      BRK    /
2617                      *
2618 1BFF        E3       MPRNT  FCB   $E3          ' MPRNT: EX (SP),HL
2619 1CØØ        EB              FCB   $EB          ' EX DE,HL
2620 1CØ1        CD              FCB   $CD          ' CALL MSX
2621 1CØ2        13              FCB   <MSX,>MSX
2622 1CØ4        EB              FCB   $EB          ' EX DE,HL
2623 1CØ5        F5              FCB   $F5          ' PUSH AF
2624 1CØ6        7E       MPRNT1 FCB   $7E          ' MPRNT1:LD A,(HL)
2625 1CØ7        23              FCB   $23          ' INC HL
2626 1CØ8        B7              FCB   $B7          ' OR A
2627 1CØ9        2Ø              FCB   $2Ø          ' JR NZ,MPRNT1
2628 1CØA        FB              FCB   MPRNT1-*-1
2629 1CØB        F1              FCB   $F1          ' POP AF
2630 1CØC        E3              FCB   $E3          ' EX (SP),HL
2631 1CØD        C9              FCB   $C9          ' RET
2632                      *
2633                      *      Output string from pointed by DE to data CR
2634                      *
2635                      *      IN     DE:String top addr.
2636                      *      OUT    /
2637                      *      BRK    /
2638                      *
2639 1CØE        F5       MSG    FCB   $F5          ' MSG: PUSH AF
2640 1CØF        3E              FCB   $3E,CR       ' LD A,CR
2641 1C11        18              FCB   $18          ' JR MSX1
2642 1C12        Ø2              FCB   MSX1-*-1
2643                      *
2644                      *      Output string from pointed by DE to data EOS
2645                      *
2646                      *      IN     DE:String top addr.
2647                      *      OUT    /
2648                      *      BRK    /
2649                      *
2650 1C13        F5       MSX    FCB   $F5          ' MSX: PUSH AF
2651 1C14        AF              FCB   $AF          ' XOR A
2652 1C15        CF       MSX1   FCB   $CF,13       ' MSX1: SWORD S$OUTLN
2653 1C17        CD              FCB   $CD          ' CALL CHKLPS
2654 1C18        1E              FCB   <CHKLPS,>CHKLPS
2655 1C1A        CF              FCB   $CF,19       ' SWORD S$OUTLPL
2656 1C1C        F1              FCB   $F1          ' POP AF
2657 1C1D        C9              FCB   $C9          ' RET
2658                      *
2659                      *      Check line printer switch
2660                      *
2661                      *      IN     /
2662                      *      OUT    Z flag:(Set:Switch off,Reset:Switch on)
2663                      *      BRK    /
2664                      *
2665 1C1E        E5       CHKLPS FCB   $E5          ' CHKLPS:PUSH HL
2666 1C1F        2A              FCB   $2A          ' LD HL,(_LPSW)
2667 1C2Ø        7C              FCB   <_LPSW,>_LPSW
2668 1C22        2C              FCB   $2C          ' INC L
2669 1C23        2D              FCB   $2D          ' DEC L
2670 1C24        E1              FCB   $E1          ' POP HL
2671 1C25        CØ              FCB   $CØ          ' RET NZ
2672 1C26        F1              FCB   $F1          ' POP AF
2673 1C27        F1              FCB   $F1          ' POP AF
2674 1C28        C9              FCB   $C9          ' RET
2675                      *
2676                      *      Output 1 charactor to line printer
2677                      *
2678                      *      IN     A:Data
2679                      *      OUT    /
2680                      *      BRK    /
2681                      *
2682                      *      When error detected,carry flag must be on
2683                      *
2684 1C29        CF       LPRNT  FCB   $CF,18       ' LPRNT: SWORD S$OUTLPC
2685 1C2B        C9              FCB   $C9          ' RET
2686                      *
2687                      *      Line printer switch off
2688                      *
2689                      *      IN     /
2690                      *      OUT    /
2691                      *      BRK    /
2692                      *
2693 1C2C        F5       LPTOF  FCB   $F5          ' LPTOF: PUSH AF
2694 1C2D        18              FCB   $18          ' JR LPTON1
2695 1C2E        Ø2              FCB   LPTON1-*-1
2696                      *
2697                      *      Line printer switch on
2698                      *
2699                      *      IN     /
2700                      *      OUT    /
2701                      *      BRK    /
2702                      *
2703 1C2F        F5       LPTON  FCB   $F5          ' LPTON: PUSH AF
2704 1C3Ø        3E              FCB   $3E          ' DB 3EH ; LD A,n
2705 1C31        AF       LPTON1 FCB   $AF          ' LPTON1:XOR A
2706 1C32        32              FCB   $32          ' LD (_LPSW),A
2707 1C33        7C              FCB   <_LPSW,>_LPSW
2708 1C35        F1              FCB   $F1          ' POP AF
2709 1C36        C9              FCB   $C9          ' RET
2710                      *
2711                      *      Input 1 line from console with echo back
2712                      *
2713                      *      IN     DE:Input buffer addr.
2714                      *      OUT    /
2715                      *      BRK    /
2716                      *
2717                      *      When abort,store $1B to input buffer top
2718                      *
2719 1C37        CF       GETL   FCB   $CF,11       ' GETL: SWORD S$INLN
2720 1C39        C9              FCB   $C9          ' RET
2721                      *
2722                      *      Cheack 'BREAK' key status
2723                      *
2724                      *      IN     /
2725                      *      OUT    Z flag:Set/Pressed,Reset/Opened
2726                      *      BRK    F
2727                      *
2728 1C3A        CF       BRKEY  FCB   $CF,9        ' BRKEY: SWORD S$CHKBRK
2729 1C3C        C9              FCB   $C9          ' RET
2730                      *
2731                      *      Input 1 charactor from key-bord real time
2732                      *
2733                      *      IN     /
2734                      *      OUT    A:Data
2735                      *      BRK    /
2736                      *
2737 1C3D        CF       GETKY  FCB   $CF,8        ' GETKY: SWORD S$INCHR
2738 1C3F        C9              FCB   $C9          ' RET
2740                      *      Input 1 charactor from key-bord with wait
2741                      *
2742                      *      IN     /
2743                      *      OUT    A:Data
2744                      *      BRK    /
2745                      *
2746             1C4Ø     FLGET  EQU   *            ' FLGET: EQU $
2747 1C4Ø        CF       INKEY  FCB   $CF,7        ' INKEY: SWORD I$INCH
2748 1C42        C9              FCB   $C9          ' RET
2749                      *
2750                      *      Read cursor location
2751                      *
2752                      *      IN     /
2753                      *      OUT    H:Cursor Y pos.
2754                      *             L:Cursor X pos.
2755                      *      BRK    HL
2756                      *
2757 1C43        CF       CSR    FCB   $CF,14       ' CSR: SWORD S$POS_XY
2758 1C45        C9              FCB   $C9          ' RET
2759                      *
2760                      *      Read charactor from console buffer
2761                      *
2762                      *      IN     H:Console Y coodinate
2763                      *             L:Console X coodinate
2764                      *      OUT    A:Console char. data
2765                      *      BRK    A
2766                      *
2767 1C46        CF       SCRN   FCB   $CF,16       ' SCRN: SWORD I$SCREAD
2768 1C48        C9              FCB   $C9          ' RET
2769                      *
2770                      *      Set cursor location
2771                      *
2772                      *      IN     H:Cursor Y pos.
2773                      *             L:Cursor X pos.
2774                      *      OUT    /
2775                      *      BRK    /
2776                      *
2777 1C49        CF       LOC    FCB   $CF,15       ' LOC: SWORD S$LOCATE
2778 1C4B        C9              FCB   $C9          ' RET
2779                      *
2780                      *      Pause by space key
2781                      *
2782                      *      IN     /
2783                      *      OUT    /
2784                      *      BRK    AF
2785                      *
2786 1C4C        CF       PAUSE  FCB   $CF,8        ' PAUSE: SWORD S$INCHR
2787 1C4E        B7              FCB   $B7          ' OR A
2788 1C4F        28              FCB   $28          ' JR Z,PAUEXT
2789 1C5Ø        ØE              FCB   PAUEXT-*-1
2790 1C51        FE              FCB   $FE,ESC      ' CP ESC
2791 1C53        28              FCB   $28          ' JR Z,PAUBRK
2792 1C54        ØF              FCB   PAUBRK-*-1
2793 1C55        FE              FCB   $FE,SPC      ' CP SPC
2794 1C57        2Ø              FCB   $2Ø          ' JR NZ,PAUEXT
2795 1C58        Ø6              FCB   PAUEXT-*-1
2796 1C59        CF       PAUWAI FCB   $CF,7        ' PAUWAI:SWORD I$INCH
2797 1C5B        FE              FCB   $FE,ESC      ' CP ESC
2798 1C5D        28              FCB   $28          ' JR Z,PAUBRK
2799 1C5E        Ø5              FCB   PAUBRK-*-1
2800 1C5F        E3       PAUEXT FCB   $E3          ' PAUEXT:EX (SP),HL
2801 1C6Ø        23              FCB   $23          ' INC HL
2802 1C61        23              FCB   $23          ' INC HL
2803 1C62        E3              FCB   $E3          ' EX (SP),HL
2804 1C63        C9              FCB   $C9          ' RET
2806 1C64        E3       PAUBRK FCB   $E3          ' PAUBRK:EX (SP),HL
2807 1C65        7E              FCB   $7E          ' LD A,(HL)
2808 1C66        23              FCB   $23          ' INC HL
2809 1C67        66              FCB   $66          ' LD H,(HL)
2810 1C68        6F              FCB   $6F          ' LD L,A
2811 1C69        E3              FCB   $E3          ' EX (SP),HL
2812 1C6A        C9              FCB   $C9          ' RET
2813                      *
2814                      *      Convert binary data to hexa-decimal in Acc.
2815                      *
2816                      *      IN     A:Binary data
2817                      *      OUT    A:Hexa-decimal data (ASCII coded)
2818                      *      BRK    AF
2819                      *
2820 1C6B        F6       ASC    FCB   $F6,$FØ      ' ASC: OR ØFØH
2821 1C6D        27              FCB   $27          ' DAA
2822 1C6E        C6              FCB   $C6,16Ø      ' ADD A,16Ø
2823 1C7Ø        CE              FCB   $CE,64       ' ADC A,64
2824 1C72        C9              FCB   $C9          ' RET
2825                      *
2826                      *      Convert hexa-decimal data to binary (4)
2827                      *
2828                      *      IN     DE:Pointer of hexa-decimal data
2829                      *      OUT    HL:Binary data
2830                      *      BRK    AF,DE(+4 inc.),HL
2831                      *
2832                      *      When error detect,carry flag must be on.
2833                      *
2834 1C73        CD       HLHEX  FCB   $CD          ' HLHEX: CALL AHEX
2835 1C74        7C              FCB   <AHEX,>AHEX
2836 1C76        67              FCB   $67          ' LD H,A
2837 1C77        D4              FCB   $D4          ' CALL NC,AHEX
2838 1C78        7C              FCB   <AHEX,>AHEX
2839 1C7A        6F              FCB   $6F          ' LD L,A
2840 1C7B        C9              FCB   $C9          ' RET
2841                      *
2842                      *      Convert hexa-decimal data to binary (2)
2843                      *
2844                      *      IN     DE:Pointer of hexa-decimal data
2845                      *      OUT    A :Binary data
2846                      *      BRK    AF,DE(+2 inc.)
2847                      *
```

▶「S-OSはピンクレディーだ！」といっていた熊本の高校生が，上京して東京の会社員になってしまいました。ほかの新入社員はほとんど大卒なのでプレッシャーも感じることがありますが，Oh！MZとともにがんばっていこうと思います。　林　聖二（18）東京都

```
2848                  *      When error detect,carry flag must be on.
2849                  *
285Ø 1C7C    C5       AHEX   FCB   $C5         ' AHEX: PUSH BC
2851 1C7D    1A              FCB   $1A         ' LD A,(DE)
2852 1C7E    13              FCB   $13         ' INC DE
2853 1C7F    CD              FCB   $CD         ' CALL HEX
2854 1C8Ø    93              FCB   <HEX,>HEX
2855 1C82    38              FCB   $38         ' JR C,AHEX1
2856 1C83    ØD              FCB   AHEX1-*-1
2857 1C84    ØF              FCB   $ØF         ' RRCA
2858 1C85    ØF              FCB   $ØF         ' RRCA
2859 1C86    ØF              FCB   $ØF         ' RRCA
286Ø 1C87    ØF              FCB   $ØF         ' RRCA
2861 1C88    4F              FCB   $4F         ' LD C,A
2862 1C89    1A              FCB   $1A         ' LD A,(DE)
2863 1C8A    13              FCB   $13         ' INC DE
2864 1C8B    CD              FCB   $CD         ' CALL HEX
2865 1C8C    93              FCB   <HEX,>HEX
2866 1C8E    38              FCB   $38         ' JR C,AHEX1
2867 1C8F    Ø1              FCB   AHEX1-*-1
2868 1C9Ø    B1              FCB   $B1         ' OR C
2869 1C91    C1       AHEX1  FCB   $C1         ' AHEX1: POP BC
287Ø 1C92    C9              FCB   $C9         ' RET
2871                  *
2872                  *      Convert hexa-decimal data to binary in Acc.
2873                  *
2874                  *      IN     A:Hexa-decimal data (ASCII coded)
2875                  *      OUT    A:Binary data
2876                  *      BRK    AF
2877                  *
2878                  *      When error detect,carry flag must be on
2879                  *
288Ø 1C93    D6       HEX    FCB   $D6,'Ø      'HEX:   SUB    "Ø"
2881 1C95    D8              FCB   $D8         ' RET C
2882 1C96    FE              FCB   $FE,1Ø      ' CP 1Ø
2883 1C98    38              FCB   $38         ' JR C,HEX1
2884 1C99    Ø7              FCB   HEX1-*-1
2885 1C9A    FE              FCB   $FE,17      ' CP 17
2886 1C9C    D8              FCB   $D8         ' RET C
2887 1C9D    D6              FCB   $D6,7       ' SUB 7
2888 1C9F    FE              FCB   $FE,$1Ø     ' CP 1ØH
2889 1CA1    3F       HEX1   FCB   $3F         ' HEX1: CCF
289Ø 1CA2    C9              FCB   $C9         ' RET
2891                  *
2892                  *      Read byte data from video memory
2893                  *
2894                  *      IN     HL:VRAM addr.
2895                  *      OUT    A :Data
2896                  *      BRK    A
2897                  *
2898 1CA3    EB       PEEK   FCB   $EB         ' PEEK: EX DE,HL
2899 1CA4    CF              FCB   $CF,22      ' SWORD S$VMREAD
29ØØ 1CA6    EB              FCB   $EB         ' EX DE,HL
29Ø1 1CA7    C9              FCB   $C9         ' RET
29Ø2                  *
29Ø3                  *      Write byte data to video memory
29Ø4                  *
29Ø5                  *      IN     HL:VRAM addr.
29Ø6                  *             A :Data
29Ø7                  *      OUT    /
29Ø8                  *      BRK    /
29Ø9                  *
291Ø 1CA8    EB       POKE   FCB   $EB         ' POKE: EX DE,HL
2911 1CA9    CF              FCB   $CF,23      ' SWORD S$VMWRIT
2912 1CAB    EB              FCB   $EB         ' EX DE,HL
2913 1CAC    C9              FCB   $C9         ' RET
2914                  *
2915                  *      Read data block from video memory
2916                  *
2917                  *      IN     BC:Byte count of data
2918                  *             DE:VRAM addr.
2919                  *             HL:Buffer addr. of main memory
292Ø                  *      OUT    /
2921                  *      BRK    AF,BC,DE,HL
2922                  *
2923 1CAD    CF       PEEK_L FCB   $CF,22      ' PEEK_L:SWORD S$VMREAD
2924 1CAF    13              FCB   $13         ' INC DE
2925 1CBØ    77              FCB   $77         ' LD (HL),A
2926 1CB1    23              FCB   $23         ' INC HL
2927 1CB2    ØB              FCB   $ØB         ' DEC BC
2928 1CB3    78       PEEK_  FCB   $78         ' PEEK_: LD A,B
2929 1CB4    B1              FCB   $B1         ' OR C
293Ø 1CB5    3Ø              FCB   $3Ø         ' JR NZ,PEEK_L
2931 1CB6    F6              FCB   PEEK_L-*-1
2932 1CB7    C9              FCB   $C9         ' RET
2933                  *
2934                  *      Write data block to video memory
2935                  *
2936                  *      IN     BC:Byte count of data
2937                  *             DE:VRAM addr.
2938                  *             HL:Buffer addr. of main memory
2939                  *      OUT    /
294Ø                  *      BRK    AF,BC,DE,HL
2941 1CB8    7E       POKE_L FCB   $7E         ' POKE_L:LD A,(HL)
2942 1CB9    23              FCB   $23         ' INC HL
2943 1CBA    CF              FCB   $CF,23      ' SWORD S$VMWRIT
2944 1CBC    13              FCB   $13         ' INC DE
2945 1CBD    ØB              FCB   $ØB         ' DEC BC
2946 1CBE    78       POKE_  FCB   $78         ' POKE_: LD A,B
2947 1CBF    B1              FCB   $B1         ' OR C
2948 1CCØ    2Ø              FCB   $2Ø         ' JR NZ,POKE_L
2949 1CC1    F6              FCB   POKE_L-*-1
295Ø 1CC2    C9              FCB   $C9         ' RET
2951                  *
2952                  *      Input from I/O port
2953                  *
2954                  *      IN     C:Port addr.
2955                  *      OUT    A:Data
2956                  *      BRK    A
2957                  *
2958 1CC3    C5       INP    FCB   $C5         ' INP: PUSH BC
2959 1CC4    Ø6              FCB   $Ø6,$FD     ' LD B,FDH
296Ø 1CC6    1A              FCB   $1A         ' LD A,(BC)
2961 1CC7    C1              FCB   $C1         ' POP BC
2962 1CC8    17              FCB   $17         ' RLA
2963 1CC9    CB1F            FDB   $CB1F       ' RR A
2964 1CCB    C9              FCB   $C9         ' RET
2965                  *
2966                  *      Output to I/O port
2967                  *
2968                  *      IN     A:data
2969                  *             C:Port addr.
297Ø                  *      OUT    /
2971                  *      BRK    /
2972                  *
2973 1CCC    C5       OUT    FCB   $C5         ' OUT: PUSH BC
2974 1CCD    Ø6              FCB   $Ø6,$FD     ' LD B,FDH
2975 1CCF    Ø2              FCB   $Ø2         ' LD (BC),A
2976 1CDØ    C1              FCB   $C1         ' POP BC
2977 1CD1    C9              FCB   $C9         ' RET
2978                  *
2979                  *      System Boot
298Ø                  *
2981                  *      IN     /
2982                  *      OUT    /
2983                  *      BRK    /
2984                  *
2985 1CD2    CF       BOOT   FCB   $CF,3Ø      ' BOOT: SWORD S$NEW_ON
2986                  *
2987                  *      Set file descriptor
2988                  *
2989                  *      IN     A :File attr.
299Ø                  *             DE:File name top addr.
2991                  *      OUT    /
2992                  *      BRK    AF,BC,DE,HL
2993                  *
2994 1CD4    CD       FILE   FCB   $CD         ' FILE: CALL FNAME
2995 1CD5    E9              FCB   <FNAME,>FNAME
2996 1CD7    D5              FCB   $D5         ' PUSH DE
2997 1CD8    21              FCB   $21         ' LD HL,NAMEBF
2998 1CD9    FA              FCB   <NAMEBF,>NAMEBF
2999 1CDB    11              FCB   $11         ' LD DE,INFBLK
3ØØØ 1CDC    ØØ              FCB   <INFBLK,>INFBLK
3ØØ1 1CDE    Ø1              FCB   $Ø1         ' LD BC,18
3ØØ2 1CDF    12              FCB   18,Ø
3ØØ3 1CE1    EDBØ            FDB   $EDBØ       ' LDIR
3ØØ4 1CE3    D1              FCB   $D1         ' POP DE
3ØØ5 1CE4    CD              FCB   $CD         ' CALL SPCUT
3ØØ6 1CE5    C1              FCB   <SPCUT,>SPCUT
3ØØ7 1CE7    B7              FCB   $B7         ' OR A
3ØØ8 1CE8    C9              FCB   $C9         ' RET

3Ø1Ø 1CE9    21       FNAME  FCB   $21         ' FNAME: LD HL,NAMEBF
3Ø11 1CEA    FA              FCB   <NAMEBF,>NAMEBF
3Ø12 1CEC    77              FCB   $77         ' LD (HL),A
3Ø13 1CED    23              FCB   $23         ' INC HL
3Ø14 1CEE    32              FCB   $32         ' LD (_FTYPE),A
3Ø15 1CEF    1F              FCB   <_FTYPE,>_FTYPE
3Ø16 1CF1    CD              FCB   $CD         ' CALL GETDEV
3Ø17 1CF2    4B              FCB   <GETDEV,>GETDEV
3Ø18 1CF4    CD              FCB   $CD         ' CALL DVCHK?
3Ø19 1CF5    15              FCB   <DVCHK@,>DVCHK@
3Ø2Ø 1CF7    D8              FCB   $D8         ' RET C
3Ø21 1CF8    32              FCB   $32         ' LD (_DSK),A
3Ø22 1CF9    5D              FCB   <_DSK,>_DSK
3Ø23 1CFB    Ø6       FILE2  FCB   $Ø6,13      ' FILE2: LD B,13
3Ø24 1CFD    CD       FILE2A FCB   $CD         ' FILE2A:CALL FILE3
3Ø25 1CFE    37              FCB   <FILE3,>FILE3
3Ø26 1DØØ    CD              FCB   $CD         ' CALL PTOSPC
3Ø27 1DØ1    8D              FCB   <PTOSPC,>PTOSPC
3Ø28 1DØ3    2Ø              FCB   $2Ø         ' JR NZ,FILE2B
3Ø29 1DØ4    Ø1              FCB   FILE2B-*-1
3Ø3Ø 1DØ5    1B              FCB   $1B         ' DEC DE
3Ø31 1DØ6    77       FILE2B FCB   $77         ' FILE2B:LD (HL),A
3Ø32 1DØ7    13              FCB   $13         ' INC DE
3Ø33 1DØ8    23              FCB   $23         ' INC HL
3Ø34 1DØ9    1Ø              FCB   $1Ø         ' DJNZ FILE2A
3Ø35 1DØA    F2              FCB   FILE2A-*-1
3Ø36 1DØB    1A              FCB   $1A         ' LD A,(DE)
3Ø37 1DØC    FE              FCB   $FE,'.      ' CP     '.'
3Ø38 1DØE    2Ø              FCB   $2Ø         ' JR NZ,FILE21
3Ø39 1DØF    Ø1              FCB   FILE21-*-1
3Ø4Ø 1D1Ø    13              FCB   $13         ' INC DE
3Ø41 1D11    Ø6       FILE21 FCB   $Ø6,3       ' FILE21:LD B,3
3Ø42 1D13    CD       FILE2C FCB   $CD         ' FILE2C:CALL FILE3
3Ø43 1D14    37              FCB   <FILE3,>FILE3
3Ø44 1D16    77              FCB   $77         ' LD (HL),A
3Ø45 1D17    13              FCB   $13         ' INC DE
3Ø46 1D18    23              FCB   $23         ' INC HL
3Ø47 1D19    1Ø              FCB   $1Ø         ' DJNZ FILE2C
3Ø48 1D1A    F8              FCB   FILE2C-*-1
3Ø49 1D1B    36              FCB   $36,SPC     ' LD (HL),SPC
3Ø5Ø 1D1D    3A              FCB   $3A         ' LD A,(_DSK)
3Ø51 1D1E    5D              FCB   <_DSK,>_DSK
3Ø52 1D2Ø    CD              FCB   $CD         ' CALL TPCHK?
3Ø53 1D21    18              FCB   <TPCHK@,>TPCHK@
3Ø54 1D23    CØ              FCB   $CØ         ' RET NZ
3Ø55 1D24    FE              FCB   $FE,'S      ' CP     'S'
3Ø56 1D26    C8              FCB   $C8         ' RET Z
3Ø57 1D27    21              FCB   $21         ' LD HL,NAMEBF+17
3Ø58 1D28    ØB              FCB   <NAME17,>NAME17
3Ø59 1D2A    Ø6              FCB   $Ø6,17      ' LD B,17
3Ø6Ø 1D2C    7E       MZØDF  FCB   $7E         ' MZØDF: LD A,(HL)
3Ø61 1D2D    FE              FCB   $FE,SPC+1   ' CP SPC+1
3Ø62 1D2F    DØ              FCB   $DØ         ' RET NC
3Ø63 1D3Ø    3E              FCB   $3E,CR      ' LD A,CR
3Ø64 1D32    77              FCB   $77         ' LD (HL),A
3Ø65 1D33    2B              FCB   $2B         ' DEC HL
3Ø66 1D34    1Ø              FCB   $1Ø         ' DJNZ MZØDF
3Ø67 1D35    F6              FCB   MZØDF-*-1
3Ø68 1D36    C9              FCB   $C9         ' RET

3Ø7Ø 1D37    D5       FILE3  FCB   $D5         ' FILE3: PUSH DE
3Ø71 1D38    CD              FCB   $CD         ' CALL SPCUT
3Ø72 1D39    C1              FCB   <SPCUT,>SPCUT
3Ø73 1D3B    1A              FCB   $1A         ' LD A,(DE)
3Ø74 1D3C    D1              FCB   $D1         ' POP DE
3Ø75 1D3D    FE              FCB   $FE,':      ' CP     ':'
3Ø76 1D3F    28              FCB   $28         ' JR Z,FILE32
3Ø77 1D4Ø    Ø6              FCB   FILE32-*-1
3Ø78 1D41    FE       FILE31 FCB   $FE,SPC     ' FILE31:CP SPC
3Ø79 1D43    38              FCB   $38         ' JR C,FILE32
3Ø8Ø 1D44    Ø2              FCB   FILE32-*-1
3Ø81 1D45    1A              FCB   $1A         ' LD A,(DE)
3Ø82 1D46    C9              FCB   $C9         ' RET

3Ø84 1D47    3E       FILE32 FCB   $3E,SPC     ' FILE32:LD A,SPC
3Ø85 1D49    1B              FCB   $1B         ' DEC DE
3Ø86 1D4A    C9              FCB   $C9         ' RET

3Ø88 1D4B    CD       GETDEV FCB   $CD         ' GETDEV:CALL SPCUT
3Ø89 1D4C    C1              FCB   <SPCUT,>SPCUT
3Ø9Ø 1D4E    13              FCB   $13         ' INC DE
3Ø91 1D4F    1A              FCB   $1A         ' LD A,(DE)
3Ø92 1D5Ø    1B              FCB   $1B         ' DEC DE
3Ø93 1D51    FE              FCB   $FE,':      ' CP     ':'
3Ø94 1D53    C2              FCB   $C2         ' JP NZ,RDVSW
3Ø95 1D54    AD              FCB   <RDVSW,>RDVSW
3Ø96 1D56    1A       GETDV1 FCB   $1A         ' GETDV1:LD A,(DE)
3Ø97 1D57    13              FCB   $13         ' INC DE
3Ø98 1D58    13              FCB   $13         ' INC DE

31ØØ 1D59    FE       TOUPER FCB   $FE,'a      'TOUPER CP     'a'
31Ø1 1D5B    D8              FCB   $D8         ' RET C
31Ø2 1D5C    FE              FCB   $FE,'z+1    ' CP     'z'+1
31Ø3 1D5E    DØ              FCB   $DØ         ' RET NC
31Ø4 1D5F    D6              FCB   $D6,SPC     ' SUB SPC
31Ø5 1D61    C9              FCB   $C9         ' RET
31Ø6                  *
31Ø7                  *      Diplay directory list
31Ø8                  *
31Ø9                  *      IN     /
311Ø                  *      OUT    /
3111                  *      BRK    AF,BC,DE,HL
3112                  *
3113 1D62    11       FPRNT  FCB   $11         ' FPRNT: LD DE,INFBLK+1
3114 1D63    Ø1              FCB   <INFBK1,>INFBK1
3115 1D65    Ø6              FCB   $Ø6,13      ' LD B,13
3116 1D67    1A       FPRNTA FCB   $1A         ' FPRNTA:LD A,(DE)
3117 1D68    CD              FCB   $CD         ' CALL FILE31
3118 1D69    41              FCB   <FILE31,>FILE31
3119 1D6B    CD              FCB   $CD         ' CALL PTOSPC
312Ø 1D6C    8D              FCB   <PTOSPC,>PTOSPC
3121 1D6E    CD              FCB   $CD         ' CALL PRINT
3122 1D6F    F5              FCB   <PRINT,>PRINT
3123 1D71    13              FCB   $13         ' INC DE
3124 1D72    1Ø              FCB   $1Ø         ' DJNZ FPRNTA
3125 1D73    F3              FCB   FPRNTA-*-1

3127 1D74    3E       FILPR1 FCB   $3E,'.      'FILPR1:LD     A,'.'
3128 1D76    CD              FCB   $CD         ' CALL PRINT
3129 1D77    F5              FCB   <PRINT,>PRINT
313Ø 1D79    Ø6              FCB   $Ø6,3       ' LD B,3
3131 1D7B    1A       FILPR2 FCB   $1A         ' FILPR2:LD A,(DE)
3132 1D7C    CD              FCB   $CD         ' CALL FILE31
3133 1D7D    41              FCB   <FILE31,>FILE31
3134 1D7F    CD              FCB   $CD         ' CALL PRINT
3135 1D8Ø    F5              FCB   <PRINT,>PRINT
3136 1D82    13              FCB   $13         ' INC DE
3137 1D83    1Ø              FCB   $1Ø         ' DJNZ FILPR2
3138 1D84    F6              FCB   FILPR2-*-1
3139 1D85    CF              FCB   $CF,8       ' SWORD S$INCHR
314Ø 1D87    FE              FCB   $FE,SPC     ' CP SPC
3141 1D89    CØ              FCB   $CØ         ' RET NZ
```


▶子供が生まれてからMZ-1500も眠ったままになっていますが，子供が大きくなったらいっしょにパソコンできればいいな，などと思いつつOh！MZを読んでいる毎日です。
正司　尚子（28）神奈川県

```
3142 1D8A      CF              FCB    $CF,7       ' SWORD S$INCH
3143 1D8C      C9              FCB    $C9         ' RET
3144                   *
3145                   *       . -> SPC
3146                   *
3147 1D8D      FE      PTOSPC  FCB    $FE,'.      ' PTOSPC CP     "."
3148 1D8F      C0              FCB    $C0         ' RET NZ
3149 1D90      3E              FCB    $3E,SPC     ' LD A,SPC
3150 1D92      C9              FCB    $C9         ' RET
3151                   *
3152                   *       Compare file name
3153                   *
3154                   *       IN     /
3155                   *       OUT    /
3156                   *       BRK    /
3157                   *
3158 1D93      E6      FSAME   FCB    $E6,%10000111 'FSAME: AND    10000111H
3159 1D95      47              FCB    $47         ' LD B,A
3160 1D96      21              FCB    $21         ' LD HL,INFBLK
3161 1D97      00              FCB    <INFBLK,>INFBLK
3162 1D99      7E              FCB    $7E         ' LD A,(HL)
3163 1D9A      E6              FCB    $E6,%10000111 ' AND    10000111H
3164 1D9C      B8              FCB    $B8         ' CP B
3165 1D9D      20              FCB    $20         ' JR NZ,FSKIP
3166 1D9E      1D              FCB    FSKIP-*-1
3167 1D9F      3A              FCB    $3A         ' LD A,(_DFDV)
3168 1DA0      20              FCB    <_DFDV,>_DFDV
3169 1DA2      F5              FCB    $F5         ' PUSH AF
3170 1DA3      3A              FCB    $3A         ' LD A,(_DSK)
3171 1DA4      5D              FCB    <_DSK,>_DSK
3172 1DA6      32              FCB    $32         ' LD (_DFDV),A
3173 1DA7      20              FCB    <_DFDV,>_DFDV
3174 1DA9      CD              FCB    $CD         ' CALL FNAME
3175 1DAA      E9              FCB    <FNAME,>FNAME
3176 1DAC      F1              FCB    $F1         ' POP AF
3177 1DAD      32              FCB    $32         ' LD (_DFDV),A
3178 1DAE      20              FCB    <_DFDV,>_DFDV
3179 1DB0      11              FCB    $11         ' LD DE,INFBLK
3180 1DB1      00              FCB    <INFBLK,>INFBLK
3181 1DB3      21              FCB    $21         ' LD HL,NAMEBF
3182 1DB4      FA              FCB    <NAMEBF,>NAMEBF
3183 1DB6      06              FCB    $06,16      ' LD B,16
3184 1DB8      CD              FCB    $CD         ' CALL TCOMP
3185 1DB9      C7              FCB    <TCOMP,>TCOMP
3186 1DBB      C0              FCB    $C0         ' RET Z

3188 1DBC      3E      FSKIP   FCB    $3E,8       ' FSKIP: LD A,8
3189 1DBE      B7              FCB    $B7         ' OR A
3190 1DBF      C9              FCB    $C9         ' RET

3192 1DC0      13      CUTLP   FCB    $13         ' CUTLP: INC DE
3193 1DC1      1A      SPCUT   FCB    $1A         ' SPCUT: LD A,(DE)
3194 1DC2      FE              FCB    $FE,SPC     ' CP SPC
3195 1DC4      28              FCB    $28         ' JR Z,CUTLP
3196 1DC5      FA              FCB    CUTLP-*-1
3197 1DC6      C9              FCB    $C9         ' RET

3199 1DC7      13      TCOMP   FCB    $13         ' TCOMP: INC DE
3200 1DC8      23              FCB    $23         ' INC HL
3201 1DC9      7E              FCB    $7E         ' LD A,(HL)
3202 1DCA      FE              FCB    $FE,SPC+1   ' CP SPC+1
3203 1DCC      30              FCB    $30         ' JR NC,TCOMP1
3204 1DCD      02              FCB    TCOMP1-*-1
3205 1DCE      AF              FCB    $AF         ' XOR A
3206 1DCF      C9              FCB    $C9         ' RET

3208 1DD0      7E      TCOMP1  FCB    $7E         ' TCOMP1:LD A,(HL)
3209 1DD1      CD              FCB    $CD         ' CALL PTOSPC
3210 1DD2      8D              FCB    <PTOSPC,>PTOSPC
3211 1DD4      4F              FCB    $4F         ' LD C,A
3212 1DD5      1A              FCB    $1A         ' LD A,(DE)
3213 1DD6      CD              FCB    $CD         ' CALL PTOSPC
3214 1DD7      8D              FCB    <PTOSPC,>PTOSPC
3215 1DD9      B9              FCB    $B9         ' CP C
3216 1DDA      C0              FCB    $C0         ' RET NZ
3217 1DDB      FE              FCB    $FE,CR      ' CP CR
3218 1DDD      C8              FCB    $C8         ' RET Z
3219 1DDE      23              FCB    $23         ' INC HL
3220 1DDF      13              FCB    $13         ' INC DE
3221 1DE0      10              FCB    $10         ' DJNZ TCOMP1
3222 1DE1      EE              FCB    TCOMP1-*-1
3223 1DE2      AF              FCB    $AF         ' XOR A
3224 1DE3      C9              FCB    $C9         ' RET
3225                   *
3226                   *       Audio cassette tape GAP
3227                   *       Future function call
3228                   *
3229 1DE4      C3      RDI     FCB    $C3         ' RDI: JP CASDMY
3230 1DE5      F6              FCB    <CASDMY,>CASDMY
3231 1DE7      C3      TROPN   FCB    $C3         ' TROPN: JP CASDMY
3232 1DE8      F6              FCB    <CASDMY,>CASDMY
3233 1DEA      C3      WRI     FCB    $C3         ' WRI: JP CASDMY
3234 1DEB      F6              FCB    <CASDMY,>CASDMY
3235 1DED      C3      TWRD    FCB    $C3         ' TWRD: JP CASDMY
3236 1DEE      F6              FCB    <CASDMY,>CASDMY
3237 1DF0      C3      TRDD    FCB    $C3         ' TRDD: JP CASDMY
3238 1DF1      F6              FCB    <CASDMY,>CASDMY
3239 1DF3      C3      TDIR    FCB    $C3         ' TDIR: JP CASDMY
3240 1DF4      F6              FCB    <CASDMY,>CASDMY
3241                   *
3242                   *       Cassette dummy GAP
3243                   *
3244 1DF6      3E      CASDMY  FCB    $3E,2       ' CASDMY:LD A,2 ; Offline
3245 1DF8      37              FCB    $37         ' SCF
3246 1DF9      C9              FCB    $C9         ' RET
3247                   *
3248                   *       File name buffer
3249                   *
3250 1DFA      0012    NAMEBF  RMB    18          ' NAMEBF:DS 18
3251           1E0B    NAME17  EQU    NAMEBF+17
3252                   *
3253                   *       Structure of stack shattle
3254                   *
3255                   *       0,Y  2   4   6   8  10 12 14 16 18 20 22
3256                   *        |   |   |   |   |   |  |  |  |  |  |  |
3257                   *       AF',AF',BC',DE',HL',AF,BC,DE,HL,IX,IY:PC
3258                   *

3260                   *
3261                   * ---- Command executer ----
3262                   *
3263 1E0C      E3      CMDGO   FCB    $E3         ' CMDGO: EX (SP),HL
3264 1E0D      F5              FCB    $F5         ' PUSH AF
3265 1E0E      7E              FCB    $7E         ' LD A,(HL)
3266 1E0F      23              FCB    $23         ' INC HL
3267 1E10      32              FCB    $32         ' LD (FUNCNO),A
3268 1E11      82              FCB    <FUNCNO,>FUNCNO
3269 1E13      F1              FCB    $F1         ' POP AF
3270 1E14      E3              FCB    $E3         ' EX (SP),HL
3271 1E15      FD              FCB    $FD,$E5     ' PUSH IY
3272 1E17      DD              FCB    $DD,$E5     ' PUSH IX
3273 1E19      E5              FCB    $E5         ' PUSH HL
3274 1E1A      D5              FCB    $D5         ' PUSH DE
3275 1E1B      C5              FCB    $C5         ' PUSH BC
3276 1E1C      F5              FCB    $F5         ' PUSH AF
3277 1E1D      08              FCB    $08         ' EX AF,AF'
3278 1E1E      D9              FCB    $D9         ' EXX
3279 1E1F      E5              FCB    $E5         ' PUSH HL
3280 1E20      D5              FCB    $D5         ' PUSH DE
3281 1E21      C5              FCB    $C5         ' PUSH BC
3282 1E22      F5              FCB    $F5         ' PUSH AF
3283 1E23      F5              FCB    $F5         ' PUSH AF
3284 1E24      ED              FCB    $ED,$73     ' LD (ZSTKPT),SP
3285 1E26      80              FCB    <ZSTKPT,>ZSTKPT
3286 1E28      AF              FCB    $AF         ' XOR A
3287 1E29      32              FCB    $32         ' LD (SBSOUT),A
3288 1E2A      05              FCB    <SBSOUT,>SBSOUT
3289 1E2C      00              FCB    $00         ' NOP
3290 1E2D      F1      CMDEXE  FCB    $F1         ' POP AF
3291 1E2E      38              FCB    $38         ' JR C,CMDILL
3292 1E2F      0F              FCB    CMDILL-*-1
3293 1E30      F1              FCB    $F1         ' POP AF
3294 1E31      C1              FCB    $C1         ' POP BC
3295 1E32      D1              FCB    $D1         ' POP DE
3296 1E33      E1              FCB    $E1         ' POP HL
3297 1E34      08              FCB    $08         ' EX AF,AF'
3298 1E35      D9              FCB    $D9         ' EXX
3299 1E36      F1              FCB    $F1         ' POP AF
3300 1E37      C1              FCB    $C1         ' POP BC
3301 1E38      D1              FCB    $D1         ' POP DE
3302 1E39      E1              FCB    $E1         ' POP HL
3303 1E3A      DD              FCB    $DD,$E1     ' POP IX
3304 1E3C      FD              FCB    $FD,$E1     ' POP IY
3305 1E3E      C9              FCB    $C9         ' RET
3306                   *
3307                   *       System call error handling routine
3308                   *
3309 1E3F      11      CMDILL  FCB    $11         ' CMDILL:LD DE,ILLMSG
3310 1E40      52              FCB    <ILLMSG,>ILLMSG
3311 1E42      CD              FCB    $CD         ' CALL MSX
3312 1E43      13              FCB    <MSX,>MSX
3313 1E45      21              FCB    $21         ' LD HL,20
3314 1E46      14              FCB    20,0
3315 1E48      39              FCB    $39         ' ADD HL,SP
3316 1E49      F9              FCB    $F9         ' LD SP,HL
3317 1E4A      E1              FCB    $E1         ' POP HL
3318 1E4B      2B              FCB    $2B         ' DEC HL
3319 1E4C      CD              FCB    $CD         ' CALL PRTHL
3320 1E4D      E4              FCB    <PRTHL,>PRTHL
3321 1E4F      C3              FCB    $C3         ' JP DEBUG
3322 1E50      80              FCB    <DEBUG,>DEBUG
3323                   *
3324                   *       System call error massage
3325                   *
3326 1E52      0D      ILLMSG  FCB    CR
3327 1E53      49              FCC    'Illigal system call detectded at $'
3328 1E75      00              FCB    EOS

3330                   *
3331                   *       End of Z-80 System module
3332                   *

3334 1E76      00E5            RMB    $1F5B-*
3335                   *
3336                   *   S-OS grobal work & system sub-rotuine hook module
3337                   *

3339                   *
3340                   *       DOS module entry equation
3341                   *
3342           234F    RDD     EQU    $234F       ' RDD EQU 234FH
3343           237C    GETFCB  EQU    $237C       ' GETFCB EQU 237CH
3344           232D    WRD     EQU    $232D       ' WRD EQU 232DH
3345           22B3    WOPEN   EQU    $22B3       ' WOPEN EQU 22B3H

3347           2100    HOT     EQU    $2100       ' HOT EQU 2100H

3349           2544    DSKRED  EQU    $2544       ' DSKRED EQU 2544H
3350           255A    DSKWRT  EQU    $255A       ' DSKWRT EQU 255AH
3351           2419    DIR     EQU    $2419       ' DIR EQU 2419H
3352           22FA    ROPEN   EQU    $22FA       ' ROPEN EQU 22FAH
3353           2508    SET     EQU    $2508       ' SET EQU 2508H
3354           2526    RESET   EQU    $2526       ' RESET EQU 2526H
3355           24AC    NAME    EQU    $24AC       ' NAME EQU 24ACH
3356           2477    KILL    EQU    $2477       ' KILL EQU 2477H

3358           25AD    RDVSW   EQU    $25AD       ' RDVSW EQU 25ADH
3359           25C9    SDVSW   EQU    $25C9       ' SDVSW EQU 25C9H

3361           286C    ERROR   EQU    $286C       ' ERROR EQU 286CH
3362                   *
3363                   * ---- S-OS Grobal work area ----
3364                   *
3365 1F5B      19      _MXLIN  FCB    25          Number of line in console
3366 1F5C      50      _WIDTH  FCB    80          Number of char. in line
3367 1F5D      41      _DSK    FCC    'A'         Current device name
3368 1F5E      0E      _FATPS  FCB    $0E,$00     FAT pos.
3369 1F60      10      _DIRPS  FCB    $10,$00     Directory pos.
3370 1F62      00      _FATBF  FCB    $00,$2E     FAT buffer addr.
3371 1F64      00      _DTBUF  FCB    $00,$2F     Data buffer addr.
3372 1F66      50      _MXTRK  FCB    80          Number of maxmum track
3373 1F67      0001    _DIRNO  RMB    1           Directory No.
3374 1F68      00      _WKSIZ  FCB    $00,$80     Size of special work area
3375 1F6A      FF      _MEMAX  FCB    $FF,$FB     Addr. of highst memory location
3376 1F6C      00      _STKAD  FCB    $00,$04     Addr. of System stack
3377 1F6E      0002    _EXADR  RMB    2           Execute addr. of file
3378 1F70      0002    _DTADR  RMB    2           Top addr. of file
3379 1F72      0002    _SIZE   RMB    2           Size of file
3380 1F74      00      _IBFAD  FCB    $00,$07     Addr. of Infomation block
3381 1F76      00      _KBFAD  FCB    $00,$06     Addr. of system input buffer
3382 1F78      84      _XYADR  FCB    $84,$00     Addr. of cursor pos. work
3383 1F7A      83      _PRCNF  FCB    $83,$00     Addr. of char. countor work
3384 1F7C      00      _LPSW   FCB    0           Line printer enable flag
3385 1F7D      00      _DVSW   FCB    0           Magnetic tape format flag
3386 1F7E      00      _USR    FCB    <HOT,>HOT   User program vector

3388 1F80      0000            RMB    $1F80-*
3389                   *
3390                   * ---- S-OS System sub-routine hook 1 ----
3391                   *
3392 1F80      E1      GETPC@  FCB    $E1         ' #GETPC:POP HL
3393 1F81      E9              FCB    $E9         ' (HL) JP (HL)

3395 1F82      000C            RMB    $1F8E-*

3397 1F8E      C3      MON@    FCB    $C3         ' #MON: JP MON
3398 1F8F      80              FCB    <DEBUG,>DEBUG
3399 1F91      00      PEEK?_  FCB    $00         ' #PEEK@:JP PEEK_
3400 1F92      B3              FCB    <PEEK_,>PEEK_
3401 1F94      C3      PEEK@   FCB    $C3         ' #PEEK: JP PEEK
3402 1F95      A3              FCB    <PEEK,>PEEK
3403 1F97      00      POKE?_  FCB    $00         ' #POKE@:JP POKE_
3404 1F98      BE              FCB    <POKE_,>POKE_
3405 1F9A      C3      POKE@   FCB    $C3         ' #POKE: JP POKE
3406 1F9B      A8              FCB    <POKE,>POKE
3407 1F9D      C3      FPRNT@  FCB    $C3         ' #FPRNT:JP FPRNT
3408 1F9E      62              FCB    <FPRNT,>FPRNT
3409 1FA0      C3      FSAME@  FCB    $C3         ' #FSAME:JP FSAME
3410 1FA1      93              FCB    <FSAME,>FSAME
3411 1FA3      C3      FILE@   FCB    $C3         ' #FILE: JP FILE
3412 1FA4      D4              FCB    <FILE,>FILE
3413 1FA6      C3      RDD@    FCB    $C3         ' #RDD: JP RDD
3414 1FA7      4F              FCB    <RDD,>RDD
3415 1FA9      C3      FCB@    FCB    $C3         ' #FCB: JP GETFCB
3416 1FAA      7C              FCB    <GETFCB,>GETFCB
3417 1FAC      C3      WRD@    FCB    $C3         ' #WRD: JP WRD
3418 1FAD      2D              FCB    <WRD,>WRD
3419 1FAF      C3      WOPEN@  FCB    $C3         ' #WOPEN:JP WOPEN
3420 1FB0      B3              FCB    <WOPEN,>WOPEN
3421 1FB2      C3      HLHEX@  FCB    $C3         ' #HLHEX:JP HLHEX
3422 1FB3      73              FCB    <HLHEX,>HLHEX
3423 1FB5      C3      AHEX@   FCB    $C3         ' #AHEX: JP AHEX
3424 1FB6      7C              FCB    <AHEX,>AHEX
3425 1FB8      C3      HEX@    FCB    $C3         ' #HEX: JP HEX
3426 1FB9      93              FCB    <HEX,>HEX
3427 1FBB      C3      ASC@    FCB    $C3         ' #ASC: JP ASC
3428 1FBC      6B              FCB    <ASC,>ASC
3429 1FBE      C3      PRTHL@  FCB    $C3         ' #PRTHL:JP PRTHL
3430 1FBF      E4              FCB    <PRTHL,>PRTHL
3431 1FC1      C3      PRTHX@  FCB    $C3         ' #PRTHX:JP PRTHX
3432 1FC2      E9              FCB    <PRTHX,>PRTHX
3433 1FC4      C3      BELL@   FCB    $C3         ' #BELL: JP BELL
3434 1FC5      CD              FCB    <BELL,>BELL
3435 1FC7      C3      PAUSE@  FCB    $C3         ' #PAUSE:JP PAUSE
```

▶突然始まった祝一平氏の「言ってはいけないことになっている真実シリーズ」。思わず笑ってうなずいてしまった。第2弾に期待します。　　中村　均（23）石川県

```
3436 1FC8       4C              FCB   <PAUSE,>PAUSE
3437 1FCA       C3      INKEY@  FCB   $C3           ' #INKEY:JP INKEY
3438 1FCB       40              FCB   <INKEY,>INKEY
3439 1FCD       C3      BRKEY@  FCB   $C3           ' #BRKEY:JP BRKEY
3440 1FCE       3A              FCB   <BRKEY,>BRKEY
3441 1FD0       C3      GETKY@  FCB   $C3           ' #GETKY:JP GETKY
3442 1FD1       3D              FCB   <GETKY,>GETKY
3443 1FD3       C3      GETL@   FCB   $C3           ' #GETL: JP GETL
3444 1FD4       37              FCB   <GETL,>GETL
3445 1FD6       C3      LPTOF@  FCB   $C3           ' #LPTOF:JP LPTOF
3446 1FD7       2C              FCB   <LPTOF,>LPTOF
3447 1FD9       C3      LPTON@  FCB   $C3           ' #LPTON:JP LPTON
3448 1FDA       2F              FCB   <LPTON,>LPTON
3449 1FDC       C3      LPRNT@  FCB   $C3           ' #LPRNT:JP LPRNT
3450 1FDD       29              FCB   <LPRNT,>LPRNT
3451 1FDF       C3      TAB@    FCB   $C3           ' #TAB: JP TAB
3452 1FE0       D2              FCB   <TAB,>TAB
3453 1FE2       C3      MPRNT@  FCB   $C3           ' #MPRNT:JP MPRNT
3454 1FE3       FF              FCB   <MPRNT,>MPRNT
3455 1FE5       C3      MSX@    FCB   $C3           ' #MSX: JP MSX
3456 1FE6       13              FCB   <MSX,>MSX
3457 1FE8       C3      MSG@    FCB   $C3           ' #MSG: JP MSG
3458 1FE9       0E              FCB   <MSG,>MSG
3459 1FEB       C3      NL@     FCB   $C3           ' #NL: JP NL
3460 1FEC       BD              FCB   <NL,>NL
3461 1FEE       C3      LTNL@   FCB   $C3           ' #LTNL: JP LTNL
3462 1FEF       C8              FCB   <LTNL,>LTNL
3463 1FF1       C3      PRNTS@  FCB   $C3           ' #PRNTS:JP PRNTS
3464 1FF2       DF              FCB   <PRNTS,>PRNTS
3465 1FF4       C3      PRINT@  FCB   $C3           ' #PRINT:JP PRINT
3466 1FF5       F5              FCB   <PRINT,>PRINT
3467 1FF7       C3      VER@    FCB   $C3           ' #VER: JP VER
3468 1FF8       9E              FCB   <VER,>VER
3469 1FFA       C3      HOT@    FCB   $C3           ' #HOT: JP HOT
3470 1FFB       00              FCB   <HOT,>HOT
3471 1FFD       C3      COLD@   FCB   $C3           ' #COLD: JP COLD
3472 1FFE       29              FCB   <COLD,>COLD
3473                    *
3474                    * ---- S-OS system sub-routine hook 2 ----
3475                    *
3476 2000       C3      DRDSB@  FCB   $C3           ' #DRDSB:JP DSKRED
3477 2001       44              FCB   <DSKRED,>DSKRED
3478 2003       C3      DWTSB@  FCB   $C3           ' #DWTSB:JP DSKWRT
3479 2004       5A              FCB   <DSKWRT,>DSKWRT
3480 2006       C3      DIR@    FCB   $C3           ' #DIR: JP DIR
3481 2007       19              FCB   <DIR,>DIR
3482 2009       C3      ROPEN@  FCB   $C3           ' #ROPEN:JP ROPEN
3483 200A       FA              FCB   <ROPEN,>ROPEN
3484 200C       C3      SET@    FCB   $C3           ' #SET: JP SET
3485 200D       08              FCB   <SET,>SET
3486 200F       C3      RESET@  FCB   $C3           ' #RESET:JP RESET
3487 2010       26              FCB   <RESET,>RESET
3488 2012       C3      NAME@   FCB   $C3           ' #NAME: JP NAME
3489 2013       AC              FCB   <NAME,>NAME
3490 2015       C3      KILL@   FCB   $C3           ' #KILL: JP KILL
3491 2016       77              FCB   <KILL,>KILL
3492 2018       C3      CSR@    FCB   $C3           ' #CSR: JP CSR
3493 2019       43              FCB   <CSR,>CSR
3494 201B       C3      SCRN@   FCB   $C3           ' #SCRN: JP SCRN
3495 201C       46              FCB   <SCRN,>SCRN
3496 201E       C3      LOC@    FCB   $C3           ' #LOC: JP LOC
3497 201F       49              FCB   <LOC,>LOC
3498 2021       C3      FLGET@  FCB   $C3           ' #FLGET:JP FLGET
3499 2022       40              FCB   <FLGET,>FLGET
3500 2024       C3      RDVSW@  FCB   $C3           ' #REVSW:JP RDVSW
3501 2025       AD              FCB   <RDVSW,>RDVSW
3502 2027       C3      SDVSW@  FCB   $C3           ' #SDVSW:JP SDVSW
3503 2028       C9              FCB   <SDVSW,>SDVSW
3504 202A       C3      INP@    FCB   $C3           ' #INP: JP INP
3505 202B       C3              FCB   <INP,>INP
3506 202D       C3      OUT@    FCB   $C3           ' #OUT: JP OUT
3507 202E       CC              FCB   <OUT,>OUT
3508 2030       C3      WIDCH@  FCB   $C3           ' #WIDCH:JP WIDCH
3509 2031       AA              FCB   <WIDCH,>WIDCH
3510 2033       C3      ERROR@  FCB   $C3           ' #ERROR:JP ERROR
3511 2034       6C              FCB   <ERROR,>ERROR
3512 2036       C3      BOOT@   FCB   $C3           ' #BOOT: JP BOOT
3513 2037       D2              FCB   <BOOT,>BOOT
3514                    *
3515                    * End of S-OS grobal work & system sub-routine hook
3516                    *
3517 2039       08C7            RMB   $2900-*
3518                    *
3519                    *       DOS module entry equation
3520                    *
3521            27E3    P_FNAM  EQU   $27E3         ' P_FNAM EQU 27E3H
3522            2851    DEVCHK  EQU   $2851         ' DEVCHK EQU 2851H
3523            2863    TPCHK   EQU   $2863         ' TPCHK EQU 2863H
3524                    *
3525                    * +++  DOS GAP  +++
3526                    *
3527 2900       C3      RDI@    FCB   $C3           ' %RDI: JP RDI
3528 2901       E4              FCB   <RDI,>RDI
3529 2903       C3      TROPN@  FCB   $C3           ' %TROPN:JP TROPN
3530 2904       E7              FCB   <TROPN,>TROPN
3531 2906       C3      WRI@    FCB   $C3           ' %WRI: JP WRI
3532 2907       EA              FCB   <WRI,>WRI
3533 2909       C3      TWRD@   FCB   $C3           ' %TWRD: JP TWRD
3534 290A       ED              FCB   <TWRD,>TWRD
3535 290C       C3      TRDD@   FCB   $C3           ' %TRDD: JP TRDD
3536 290D       F0              FCB   <TRDD,>TRDD
3537 290F       C3      TDIR@   FCB   $C3           ' %TDIR: JP TDIR
3538 2910       F3              FCB   <TDIR,>TDIR
3539 2912       C3      P?FNAM  FCB   $C3           ' P%FNAM:JP P_FNAM
3540 2913       E3              FCB   <P_FNAM,>P_FNAM
3541 2915       C3      DVCHK@  FCB   $C3           ' %DEVCHK:JP DEVCHK
3542 2916       51              FCB   <DEVCHK,>DEVCHK
3543 2918       C3      TPCHK@  FCB   $C3           ' %TPCHK:JP TPCHK
3544 2919       63              FCB   <TPCHK,>TPCHK

3546 291B       0003            RMB   3             ' DS 3

3548 291E       0001    _OPNFG  RMB   1             ' %OPNFG:DS 1
3549 291F       0001    _FTYPE  RMB   1             ' %FTYPE:DS 1
3550 2920       41      _DFDV   FCC   'A'           ' %DFDV: DM     "A"

3552 2921       0009            RMB   9             ' DS 9

3554 292A       E5      PARSC@  FCB   $E5           ' %PARSC:PUSH HL
3555 292B       2A              FCB   $2A           ' LD HL,(SIZE@@)
3556 292C       12              FCB   <SIZE@@,>SIZE@@
3557 292E       22              FCB   $22           ' LD (_SIZE),HL
3558 292F       72              FCB   <_SIZE,>_SIZE
3559 2931       2A              FCB   $2A           ' LD HL,(DTADR?)
3560 2932       14              FCB   <DTADR@,>DTADR@
3561 2934       22              FCB   $22           ' LD (_DTADR),HL
3562 2935       70              FCB   <_DTADR,>_DTADR
3563 2937       2A              FCB   $2A           ' LD HL,(EXADR?)
3564 2938       16              FCB   <EXADR@,>EXADR@
3565 293A       22              FCB   $22           ' LD (_EXADR),HL
3566 293B       6E              FCB   <_EXADR,>_EXADR
3567 293D       E1              FCB   $E1           ' POP HL
3568 293E       C9              FCB   $C9           ' RET

3570 293F       E5      PARCS@  FCB   $E5           ' %PARCS:PUSH HL
3571 2940       2A              FCB   $2A           ' LD HL,(_SIZE)
3572 2941       72              FCB   <_SIZE,>_SIZE
3573 2943       22              FCB   $22           ' LD (SIZE@@),HL
3574 2944       12              FCB   <SIZE@@,>SIZE@@
3575 2946       2A              FCB   $2A           ' LD HL,(_DTADR)
3576 2947       70              FCB   <_DTADR,>_DTADR
3577 2949       22              FCB   $22           ' LD (DTADR?),HL
3578 294A       14              FCB   <DTADR@,>DTADR@
3579 294C       2A              FCB   $2A           ' LD HL,(_EXADR)
3580 294D       6E              FCB   <_EXADR,>_EXADR
3581 294F       22              FCB   $22           ' LD (EXADR?),HL
3582 2950       16              FCB   <EXADR@,>EXADR@
3583 2952       E1              FCB   $E1           ' POP HL
3584 2953       C9              FCB   $C9           ' RET
3585                    *
3586                    * +++  End of DOS GAP  +++
3587                    *

3589                    *
3590                    * +++  Auxaly memory device control module  +++
3591                    *
3592 2954       01AC            RMB   $2B00-*
3593                    *
3594                    *       Module hook & work area
3595                    *
3596 2B00       C3              FCB   $C3           ' JP DRD
3597 2B01       07              FCB   <DRD,>DRD
3598 2B03       C3              FCB   $C3           ' JP DWRT
3599 2B04       10              FCB   <DWRT,>DWRT

3601 2B06       00      UNITNO  FCB   0             ' UNITNO:DB 0
3602                    *
3603                    *       Read floppy disk
3604                    *
3605                    *       IN     A :Number of access record
3606                    *              DE:Number of top record
3607                    *              HL:Read buffer addr.
3608                    *       OUT    A :Error code
3609                    *       BRK    A
3610                    *
3611 2B07       C5      DRD     FCB   $C5           ' DRD: PUSH BC
3612 2B08       ED4B            FDB   $ED4B         ' LD BC,(UNITNO)
3613 2B0A       06              FCB   <UNITNO,>UNITNO
3614 2B0C       CF              FCB   $CF,26        ' SWORD S$FDREAD
3615 2B0E       C1              FCB   $C1           ' POP BC
3616 2B0F       C9              FCB   $C9           ' RET
3617                    *
3618                    *       Write floppy disk
3619                    *
3620                    *       IN     A :Number of access record
3621                    *              DE:Number of top record
3622                    *              HL:Write buffer addr.
3623                    *       OUT    A :Error code
3624                    *       BRK    A
3625                    *
3626 2B10       C5      DWRT    FCB   $C5           ' DWRT: PUSH BC
3627 2B11       ED4B            FDB   $ED4B         ' LD BC,(UNITNO)
3628 2B13       06              FCB   <UNITNO,>UNITNO
3629 2B15       CF              FCB   $CF,27        ' SWORD S$FDWRIT
3630 2B17       C1              FCB   $C1           ' POP BC
3631 2B18       C9              FCB   $C9           ' RET
3632                    *
3633                    *       Audio cassette handling routine
3634                    *
3635                    *       not coding yet , future
3636                    *

3638            2B18.   PROEND  EQU   *-1
```

リスト12 エミュレータソースリスト

```
0                       *                                   *
1                       * +++  Z-80 emulator module  +++    *
2                       *                                   *

4                       *    LABEL EQUATION

6               1E2D    CMDEXE EQU    $1E2D
7               0080    ZSTKPT EQU    $0080
8               0082    FUNCNO EQU    $0082

10 0000         1080           RMB    $1080-*
11              10             SETDP  $10

13                      *             Work Area

15 1080         0001    DATA    RMB   1            Fetched Code
16 1081         0001    DATA2   RMB   1            b5~b3
17 1082         0001    DATA3   RMB   1            b2~b0

19                      *             Registers  and Work

21              1083    REGTOP  EQU   *
22 1083         0000    B1      FDB   $0000        BC
23 1085         0000    D1      FDB   $0000        DE
24 1087         0000    H1      FDB   $0000        HL
25 1089         0001    WORK1   RMB   1
26 108A         00      A1      FCB   $00          A
27 108B         00      F1      FCB   $00          F

29 108C         0000    B2      FDB   $0000        BC'
30 108E         0000    D2      FDB   $0000        DE'
31 1090         0000    H2      FDB   $0000        HL'
32 1092         00      INDEX   FCB   0            Index Flag
33 1093         00      A2      FCB   $00          A'
34 1094         00      F2      FCB   $00          F'

36 1095         0000    IX      FDB   $0000        IX
37 1097         0000    IY      FDB   $0000        IY
38 1099         0400    SP      FDB   $0400        SP
39 109B         0000    PC      FDB   $0000        PC

41                      *##########################

43              109D    INDEX2  EQU   *            (HL) <==>  (IX+d) or (IY+d)
44 109D    D6 92                LDB   INDEX
45 109F    27 0B                BEQ   mHL
46 10A1    DE 9B                LDU   PC
47 10A3    E6 C0                LDB   ,U+          Get Displacement
48 10A5    DF 9B                STU   PC
49 10A7    DE 87                LDU   H1
50 10A9    33 C5                LEAU  B,U
51 10AB    39                   RTS

53              10AC    mHL     EQU   *
54 10AC    DE 87                LDU   H1
55 10AE    39                   RTS

57              10AF    INDEX3  EQU   *            (HL) <==> (IX+d) or (IY+d)
58 10AF    D6 92                LDB   INDEX
59 10B1    27 F9                BEQ   mHL
60 10B3    2B 0B                BMI   mIYd
61 10B5    DE 9B        mIXd    LDU   PC
```


▶国道沿いをはじめ，もっぱら郊外に雨後の筍のごとくビリヤード場ができています。4つ玉のできるところが少ないのが残念。ポケットもいいけど基本はキャロムだぜい。

松葉　圭介（22）石川県

```
 62 10B7    E6 CØ           LDB   ,U+
 63 10B9    DF 9B           STU   PC
 64 10BB    DE 95           LDU   IX
 65 10BD    33 C5           LEAU  B,U
 66 10BF    39              RTS

 68 10CØ    DE 9B    mIYd   LDU   PC
 69 10C2    E6 CØ           LDB   ,U+
 7Ø 10C4    DF 9B           STU   PC
 71 10C6    DE 97           LDU   IY
 72 10C8    33 C5           LEAU  B,U
 73 10CA    39              RTS

 75 10CB    9E 9B    GETDTA LDX   PC          Get Data,Data1,2,3
 76 10CD    A6 8Ø           LDA   ,X+
 77 10CF    9F 9B           STX   PC          Store ProgramCounter
 78 10D1    97 8Ø           STA   DATA
 79 10D3    84 Ø7           ANDA  #%ØØØØØ111 A=ØØØØØ b2 b1 bØ
 8Ø 10D5    97 82           STA   DATA3
 81 10D7    96 8Ø           LDA   DATA
 82 10D9    44              LSRA
 83 10DA    44              LSRA
 84 10DB    44              LSRA
 85 10DC    84 Ø7           ANDA  #%ØØØØØ111 A=ØØØØØ b5 b4 b3
 86 10DE    97 81           STA   DATA2
 87 10EØ    96 8Ø           LDA   DATA
 88 10E2    5F              CLRB
 89 10E3    48              ASLA
 9Ø 10E4    59              ROLB
 91 10E5    48              ASLA
 92 10E6    59              ROLB              B=ØØØØØ b7 b6 Ø
 93 10E7    58              ASLB
 94 10E8    39              RTS

 96                  *##########################

 98            10E9  Z8Ø    EQU   *
 99 10E9    CC 1E2D         LDD   #CMDEXE     Return from 68Ø9
1ØØ 10EC    FD 1Ø9B         STD   >PC
1Ø1 10EF    2Ø Ø6           BRA   ENTEMU

1Ø3            1ØF1  PCØØØØ EQU   *           Reset Z8Ø
1Ø4 1ØF1    CC ØØØØ         LDD   #$ØØØØ
1Ø5 1ØF4    FD 1Ø9B         STD   >PC
1Ø6 1ØF7    34 Ø8    ENTEMU PSHS  DP
1Ø7 1ØF9    86 1Ø           LDA   #$1Ø
1Ø8 1ØFB    1F 8B           TFR   A,DP
1Ø9 1ØFD 1Ø8E 1Ø83          LDY   #REGTOP
11Ø 11Ø1    ØF 92           CLR   INDEX
111            11Ø3  NEXT   EQU   *
112 11Ø3    9D CB           JSR   GETDTA
113 11Ø5    8E 11ØE         LDX   #JPTBL
114 11Ø8    AE 85           LDX   B,X
115 11ØA    AD 84           JSR   ,X
116 11ØC    2Ø F5           BRA   NEXT

118            11ØE  JPTBL  EQU   *
119 11ØE       1116         FDB   COLØ        ØØH~3FH
12Ø 111Ø       144Ø         FDB   COL1        4ØH~7FH
121 1112       1465         FDB   COL2        8ØH~ØBFH
122 1114       1543         FDB   COL3        ØCØH~ØFFH

124                  *=====================================

126            1116  COLØ   EQU   *           $ØØ~$3F
127 1116    96 8Ø           LDA   DATA
128 1118    84 ØF           ANDA  #$ØF
129 111A    48              ASLA              A=A*2
13Ø 111B    8E 1122         LDX   #JPTBL2
131 111E    AE 86           LDX   A,X
132 112Ø    6E 84           JMP   ,X

134            1122  JPTBL2 EQU   *
135 1122       1142         FDB   COMØ_Ø
136 1124       1199         FDB   COMØ_1
137 1126       11D8         FDB   COMØ_2
138 1128       1234         FDB   COMØ_3
139 112A       1266         FDB   COMØ_4
14Ø 112C       1297         FDB   COMØ_5
141 112E       12CC         FDB   COMØ_6
142 113Ø       12E6         FDB   COMØ_7
143 1132       1142         FDB   COMØ_8
144 1134       11B7         FDB   COMØ_9
145 1136       12Ø5         FDB   COMØ_A
146 1138       124C         FDB   COMØ_B
147 113A       1266         FDB   COMØ_C
148 113C       1297         FDB   COMØ_D
149 113E       12CC         FDB   COMØ_E
15Ø 114Ø       12E6         FDB   COMØ_F

152                  *-----------------------------------------------------

154            1142  COMØ_Ø EQU   *
155            1142  COMØ_8 EQU   *
156 1142    96 81           LDA   DATA2       Get b5 b4 b3
157 1144    48              ASLA              A=A*2
158 1145    8E 114C         LDX   #JTBØ_Ø
159 1148    AE 86           LDX   A,X
16Ø 114A    6E 84           JMP   ,X

162            114C  JTBØ_Ø EQU   *
163 114C       115C         FDB   NOP
164 114E       119Ø         FDB   EXAFAF
165 115Ø       115D         FDB   DJNZ
166 1152       1161         FDB   JR
167 1154       116A         FDB   JRNZ
168 1156       1172         FDB   JRZ
169 1158       117A         FDB   JRNC
17Ø 115A       1182         FDB   JRC

172            115C  NOP    EQU   *
173            115C  EI     EQU   *
174            115C  DI     EQU   *
175            115C  PORT   EQU   *
176            115C  IMØ12  EQU   *
177 115C    39              RTS

179            115D  DJNZ   EQU   *
18Ø 115D    ØA 83           DEC   B1
181 115F    27 27           BEQ   SKIP1
182 1161    9E 9B    JR     LDX   PC          JR e
183 1163    A6 8Ø           LDA   ,X+
184 1165    3Ø 86           LEAX  A,X
185 1167    9F 9B           STX   PC
186 1169    39              RTS

188            116A  JRNZ   EQU   *
189 116A    96 8B           LDA   F1
19Ø 116C    85 4Ø           BITA  #%Ø1ØØØØØØ
191 116E    27 F1           BEQ   JR          If Z(flag)=Ø
192 117Ø    2Ø 16           BRA   SKIP1

194            1172  JRZ    EQU   *
195 1172    96 8B           LDA   F1
196 1174    85 4Ø           BITA  #%Ø1ØØØØØØ
197 1176    26 E9           BNE   JR          If Z(flag)=1
198 1178    2Ø ØE           BRA   SKIP1

2ØØ            117A  JRNC   EQU   *
2Ø1 117A    96 8B           LDA   F1
2Ø2 117C    85 Ø1           BITA  #%ØØØØØØØ1
2Ø3 117E    27 E1           BEQ   JR          If C(flag)=Ø
2Ø4 118Ø    2Ø Ø6           BRA   SKIP1

2Ø6            1182  JRC    EQU   *
2Ø7 1182    96 8B           LDA   F1
2Ø8 1184    85 Ø1           BITA  #%ØØØØØØØ1
2Ø9 1186    26 D9           BNE   JR          If C(flag)=1

211            1188  INnA   EQU   *           IN A,(n)
212            1188  OUTAn  EQU   *           OUT (n),A
213            1188  SKIP1  EQU   *           1 Byte Skip
214 1188    DC 9B           LDD   PC
215 118A    C3 ØØØ1         ADDD  #1
216 118D    DD 9B           STD   PC
217 118F    39              RTS

219            119Ø  EXAFAF EQU   *
22Ø 119Ø    DC 8A           LDD   A1          Get AF
221 1192    9E 93           LDX   A2          Get AF'
222 1194    DD 93           STD   A2
223 1196    9F 8A           STX   A1
224 1198    39              RTS

226                  *-----------------------------------------------------

228            1199  COMØ_1 EQU   *           LD rp,nn
229 1199    96 81           LDA   DATA2       Get b5 b4 b3 (b3=Ø)
23Ø 119B    81 Ø6           CMPA  #$Ø6
231 119D    27 ØD           BEQ   LDSPnn
232 119F    33 A6           LEAU  A,Y
233 11A1    9E 9B           LDX   PC
234 11A3    E6 8Ø           LDB   ,X+
235 11A5    A6 8Ø           LDA   ,X+         Get nn
236 11A7    9F 9B           STX   PC
237 11A9    ED C4           STD   ,U          Store nn to Register
238 11AB    39              RTS

24Ø            11AC  LDSPnn EQU   *
241 11AC    9E 9B           LDX   PC
242 11AE    EC 81           LDD   ,X++        Get nn
243 11BØ    9F 9B           STX   PC
244 11B2    D7 99           STB   SP          Store nn to SP
245 11B4    97 9A           STA   SP+1
246 11B6    39              RTS

248                  *-------------------------

25Ø            11B7  COMØ_9 EQU   *           ADD HL,rp
251 11B7    96 81           LDA   DATA2
252 11B9    84 Ø6           ANDA  #%ØØØØØ11Ø
253 11BB    81 Ø6           CMPA  #$Ø6
254 11BD    27 11           BEQ   ADHLSP

256 11BF    EC A6           LDD   A,Y         Get rp
257 11C1    D3 87           ADDD  H1          ADD HL,rp
258 11C3    DD 87           STD   H1          (Carry is not move)
259 11C5    96 8B    ADHLr1 LDA   F1          Get Flag (Carry is not move)
26Ø 11C7    84 C4           ANDA  #%11ØØØ1ØØ (Carry is not move)
261 11C9    24 Ø2           BCC   NOTC4
262 11CB    8A Ø1           ORA   #%ØØØØØØØ1
263 11CD    97 8B    NOTC4  STA   F1          Store Flag
264 11CF    39              RTS

266            11DØ  ADHLSP EQU   *           ADD HL,SP
267 11DØ    DC 99           LDD   SP
268 11D2    D3 87           ADDD  H1
269 11D4    DD 87           STD   H1
27Ø 11D6    2Ø ED           BRA   ADHLr1

272                  *-----------------------------------------------------

274            11D8  COMØ_2 EQU   *
275 11D8    96 81           LDA   DATA2
276 11DA    81 Ø4           CMPA  #$Ø4
277 11DC    24 Ø5           BCC   LDmnn?      If A=4 or A=6
278                  *                        LD (rp),A
279 11DE    D6 8A           LDB   A1          Get DATA
28Ø 11EØ    E7 B6           STB   [A,Y]       Store DATA
281 11E2    39              RTS

283            11E3  LDmnn? EQU   *
284 11E3    26 11           BNE   LDmnnA      If A=6
285                  *                        LD (nn),HL
286 11E5    9E 9B           LDX   PC
287 11E7    E6 8Ø           LDB   ,X+         Get nn
288 11E9    A6 8Ø           LDA   ,X+
289 11EB    9F 9B           STX   PC
29Ø 11ED    1F Ø1           TFR   D,X
291 11EF    DC 87           LDD   H1          Get HL
292 11F1    E7 84           STB   ,X          Store HL to (nn)
293 11F3    A7 Ø1           STA   1,X
294 11F5    39              RTS

296 11F6    9E 9B    LDmnnA LDX   PC          LD (nn),A
297 11F8    E6 8Ø           LDB   ,X+         Get nn
298 11FA    A6 8Ø           LDA   ,X+
299 11FC    9F 9B           STX   PC
3ØØ 11FE    1F Ø1           TFR   D,X
3Ø1 12ØØ    96 8A           LDA   A1
3Ø2 12Ø2    A7 84           STA   ,X          Store A to (nn)
3Ø3 12Ø4    39              RTS

3Ø5                  *-------------------------

3Ø7            12Ø5  COMØ_A EQU   *
3Ø8 12Ø5    96 81           LDA   DATA2
3Ø9 12Ø7    84 Ø6           ANDA  #%ØØØØØ11Ø Get b5 b4 b3 (b3=Ø)
31Ø 12Ø9    81 Ø4           CMPA  #$Ø4
311 12ØB    24 Ø5           BCC   LD?mnn      If A=4 or A=6
312                  *                        LD A,(rp) rp=BC,DE
313 12ØD    E6 B6           LDB   [A,Y]       Get (rp)
314 12ØF    D7 8A           STB   A1
315 1211    39              RTS

317            1212  LD?mnn EQU   *
318 1212    26 11           BNE   LDAmnn      If A=6 GOTO LD A,(nn)
319 1214    9E 9B           LDX   PC          LD HL,(nn)
32Ø 1216    E6 8Ø           LDB   ,X+         Get nn
321 1218    A6 8Ø           LDA   ,X+
322 121A    9F 9B           STX   PC
323 121C    1F Ø1           TFR   D,X
324 121E    E6 84           LDB   ,X          Get (nn)
325 122Ø    A6 Ø1           LDA   1,X
326 1222    DD 87           STD   H1          Store to HL
327 1224    39              RTS

329            1225  LDAmnn EQU   *           LD A,(nn)
33Ø 1225    9E 9B           LDX   PC
331 1227    E6 8Ø           LDB   ,X+
332 1229    A6 8Ø           LDA   ,X+
333 122B    9F 9B           STX   PC
334 122D    1F Ø1           TFR   D,X
335 122F    A6 84           LDA   ,X
336 1231    97 8A           STA   A1          Store to Acc
337 1233    39              RTS

339                  *-----------------------------------------------------

341            1234  COMØ_3 EQU   *
342 1234    96 81           LDA   DATA2
343 1236    81 Ø6           CMPA  #6
344 1238    27 ØA           BEQ   INCSP
345                  *                        INC rp
346 123A    33 A6           LEAU  A,Y
347 123C    EC C4           LDD   ,U          Get rp
348 123E    C3 ØØØ1         ADDD  #1
349 1241    ED C4           STD   ,U          Store rp
35Ø 1243    39              RTS

352            1244  INCSP  EQU   *           INC SP
353 1244    DC 99           LDD   SP
354 1246    C3 ØØØ1         ADDD  #1
355 1249    DD 99           STD   SP
```

▶中２の弟に彼女ができたらしい。まだ僕にはいないのに……。やはりパソコンに夢中なのがいけないのかな。　　木村　明（15）神奈川県

```
356 124B   39                 RTS

358                    *-------------------------

360            124C   COMØ_B EQU   *
361 124C   96 81              LDA   DATA2
362 124E   84 Ø6              ANDA  #%ØØØØØ11Ø Get b5 b4 b3 (b3=Ø)
363 125Ø   81 Ø6              CMPA  #6
364 1252   27 ØA              BEQ   DECSP
365                    *                  DEC rp
366 1254   33 A6              LEAU  A,Y
367 1256   EC C4              LDD   ,U
368 1258   83 ØØØ1            SUBD  #1
369 125B   ED C4              STD   ,U
37Ø 125D   39                 RTS

372            125E   DECSP  EQU   *
373 125E   DC 99              LDD   SP
374 126Ø   83 ØØØ1            SUBD  #1
375 1263   DD 99              STD   SP
376 1265   39                 RTS

378                    *-------------------------------------------------

38Ø            1266   COMØ_4 EQU   *
381            1266   COMØ_C EQU   *
382 1266   96 81              LDA   DATA2
383 1268   81 Ø6              CMPA  #6
384 126A   27 22              BEQ   INCmHL
385                    *                  INC r
386 126C   E6 A6      INCr    LDB   A,Y         Get r (r=B,C,D,E,H,L,A)
387 126E   5C                 INCB
388 126F   E7 A6              STB   A,Y
389 1271   96 8B      FLGINC  LDA   F1          Get Flag
39Ø 1273   84 Ø1              ANDA  #%ØØØØØØØ1 Keep Carry
391 1275   C1 8Ø              CMPB  #$8Ø
392 1277   26 Ø2              BNE   NTPV1
393 1279   8A Ø4              ORA   #%ØØØØØ1ØØ Set P/V (OVER FLOW)
394 127B   5D         NTPV1   TSTB
395 127C   26 Ø3              BNE   NOTZ1
396 127E   8A 4Ø              ORA   #%Ø1ØØØØØØ Set Zero
397 128Ø   5D                 TSTB
398 1281   2A Ø2      NOTZ1   BPL   NOTS1
399 1283   8A 8Ø              ORA   #%1ØØØØØØØ Set Sign
4ØØ 1285   C4 ØF      NOTS1   ANDB  #$ØF
4Ø1 1287   26 Ø2              BNE   NOTH1
4Ø2 1289   8A 1Ø              ORA   #%ØØØ1ØØØØ Set Half Carry
4Ø3 128B   97 8B      NOTH1   STA   F1
4Ø4 128D   39                 RTS

4Ø6            128E   INCmHL EQU   *           INC (HL)
4Ø7 128E   9D 9D              JSR   INDEX2
4Ø8 129Ø   E6 C4              LDB   ,U          Get (HL)
4Ø9 1292   5C                 INCB
41Ø 1293   E7 C4              STB   ,U
411 1295   2Ø DA              BRA   FLGINC

413                    *-------------------------

415            1297   COMØ_5 EQU   *
416            1297   COMØ_D EQU   *
417 1297   96 81              LDA   DATA2
418 1299   81 Ø6              CMPA  #$Ø6
419 129B   27 26              BEQ   DECmHL
42Ø 129D   E6 A6      DECr    LDB   A,Y         DEC r
421 129F   5A                 DECB
422 12AØ   E7 A6              STB   A,Y
423 12A2   96 8B      FLGDEC  LDA   F1
424 12A4   84 Ø1              ANDA  #%ØØØØØØØ1 Keep Carry
425 12A6   8A Ø2              ORA   #%ØØØØØØ1Ø Set N
426 12A8   C1 7F              CMPB  #$7F
427 12AA   26 Ø2              BNE   NTPV2
428 12AC   8A Ø4              ORA   #%ØØØØØ1ØØ Set P/V
429 12AE   5D         NTPV2   TSTB
43Ø 12AF   26 Ø3              BNE   NOTZ2
431 12B1   8A 4Ø              ORA   #%Ø1ØØØØØØ Set Zero
432 12B3   5D                 TSTB
433 12B4   2A Ø2      NOTZ2   BPL   NOTS2
434 12B6   8A 8Ø              ORA   #%1ØØØØØØØ Set Sign
435 12B8   C4 ØF      NOTS2   ANDB  #$ØF
436 12BA   C1 ØF              CMPB  #$ØF
437 12BC   26 Ø2              BNE   NOTH2
438 12BE   8A 1Ø              ORA   #%ØØØ1ØØØØ Set HalfCarry
439 12CØ   97 8B      NOTH2   STA   F1
44Ø 12C2   39                 RTS

442            12C3   DECmHL EQU   *           DEC (HL)
443 12C3   9D 9D              JSR   INDEX2
444 12C5   E6 C4              LDB   ,U          Get (HL)
445 12C7   5A                 DECB
446 12C8   E7 C4              STB   ,U
447 12CA   2Ø D6              BRA   FLGDEC

449                    *-------------------------------------------------

451            12CC   COMØ_6 EQU   *
452            12CC   COMØ_E EQU   *
453 12CC   D6 81              LDB   DATA2
454 12CE   C1 Ø6              CMPB  #$6
455 12DØ   27 Ø9              BEQ   LDmHLn
456                    *                  LD r,n
457            12D2   LDrn   EQU   *
458 12D2   9E 9B              LDX   PC
459 12D4   A6 8Ø              LDA   ,X+         Get n
46Ø 12D6   9F 9B              STX   PC
461 12D8   A7 A5              STA   B,Y         Store r
462 12DA   39                 RTS

464            12DB   LDmHLn EQU   *           LD (HL),n
465 12DB   9D 9D              JSR   INDEX2
466 12DD   9E 9B              LDX   PC
467 12DF   A6 8Ø              LDA   ,X+         Get n
468 12E1   9F 9B              STX   PC
469 12E3   A7 C4              STA   ,U          Store Data to (HL)
47Ø 12E5   39                 RTS

472                    *-------------------------------------------------

474            12E6   COMØ_7 EQU   *
475            12E6   COMØ_F EQU   *
476 12E6   96 81              LDA   DATA2       Get b5 b4 b3
477 12E8   48                 ASLA              A=A*2
478 12E9   8E 12FØ            LDX   #JTBØ_7
479 12EC   AE 86              LDX   A,X
48Ø 12EE   6E 84              JMP   ,X

482            12FØ   JTBØ_7 EQU   *
483 12FØ       13ØØ           FDB   RLCA
484 12F2       1322           FDB   RRCA
485 12F4       1336           FDB   RLA
486 12F6       134B           FDB   RRA
487 12F8       1374           FDB   DAA
488 12FA       1425           FDB   CPL
489 12FC       141C           FDB   SCF
49Ø 12FE       142E           FDB   CCF

492            13ØØ   RLCA   EQU   *
493 13ØØ   96 8A              LDA   A1          Get Acc
494 13Ø2   8D Ø9              BSR   RLCs
495 13Ø4   97 8A              STA   A1
496 13Ø6   39                 RTS

498            13Ø7   RLC    EQU   *
499 13Ø7   96 89              LDA   WORK1
5ØØ 13Ø9   8D Ø2              BSR   RLCs
5Ø1 13ØB   2Ø 4F              BRA   STORE1

5Ø3            13ØD   RLCs   EQU   *
5Ø4 13ØD   48                 ASLA
5Ø5 13ØE   25 Ø7              BCS   CARRY1
5Ø6            131Ø   NCARY1 EQU   *
5Ø7 131Ø   D6 8B              LDB   F1          Get Flag
5Ø8 1312   C4 C4              ANDB  #%11ØØØ1ØØ
5Ø9 1314   D7 8B              STB   F1          Store Flag
51Ø 1316   39                 RTS
511            1317   CARRY1 EQU   *
512 1317   8A Ø1              ORA   #%ØØØØØØØ1 Set bØ
513            1319   CARY11 EQU   *
514 1319   D6 8B              LDB   F1          Get Flag
515 131B   C4 C4              ANDB  #%11ØØØ1ØØ
516 131D   CA Ø1              ORB   #%ØØØØØØØ1 Set CARRY
517 131F   D7 8B              STB   F1          Store Flag
518 1321   39                 RTS

52Ø            1322   RRCA   EQU   *
521 1322   96 8A              LDA   A1          Get Acc
522 1324   8D Ø9              BSR   RRCs
523 1326   97 8A              STA   A1
524 1328   39                 RTS

526            1329   RRC    EQU   *
527 1329   96 89              LDA   WORK1
528 132B   8D Ø2              BSR   RRCs
529 132D   2Ø 2D              BRA   STORE1

531            132F   RRCs   EQU   *
532 132F   44                 LSRA
533 133Ø   24 DE              BCC   NCARY1
534 1332   8A 8Ø              ORA   #%1ØØØØØØØ Set b7
535 1334   2Ø E3              BRA   CARY11

537            1336   RLA    EQU   *
538 1336   96 8A              LDA   A1
539 1338   8D Ø9              BSR   RLs
54Ø 133A   97 8A              STA   A1
541 133C   39                 RTS

543            133D   RL     EQU   *
544 133D   96 89              LDA   WORK1
545 133F   8D Ø2              BSR   RLs
546 1341   2Ø 19              BRA   STORE1

548 1343   D6 8B      RLs     LDB   F1
549 1345   54                 LSRB              Get Carry
55Ø 1346   49                 ROLA
551            1347   RLs1   EQU   *
552            1347   RRs1   EQU   *
553 1347   25 DØ              BCS   CARY11
554 1349   2Ø C5              BRA   NCARY1

556            134B   RRA    EQU   *
557 134B   96 8A              LDA   A1
558 134D   8D Ø3              BSR   RRs
559 134F   97 8A              STA   A1
56Ø 1351   39                 RTS

562 1352   D6 8B      RRs     LDB   F1
563 1354   54                 LSRB              Get Carry
564 1355   46                 RORA
565 1356   2Ø EF              BRA   RRs1

567            1358   RR     EQU   *
568 1358   96 89              LDA   WORK1
569 135A   8D F6              BSR   RRs

571            135C   STORE1 EQU   *
572 135C   D6 82              LDB   DATA3
573 135E   C1 Ø6              CMPB  #$Ø6
574 136Ø   26 Ø9              BNE   STr1
575 1362   A7 C4              STA   ,U
576 1364   D6 8B              LDB   F1
577 1366   C4 Ø1              ANDB  #%ØØØØØØØ1 Carry is already Set
578 1368   4D                 TSTA
579 1369   2Ø 69              BRA   NOTC3
58Ø            136B   STr1   EQU   *
581 136B   A7 A5              STA   B,Y
582 136D   D6 8B              LDB   F1
583 136F   C4 Ø1              ANDB  #%ØØØØØØØ1 Carry is already Set
584 1371   4D                 TSTA
585 1372   2Ø 6Ø              BRA   NOTC3

587            1374   DAA    EQU   *
588 1374   96 8A              LDA   A1          Get Acc
589 1376   D6 8B              LDB   F1          Get Flag
59Ø 1378   C5 Ø2              BITB  #%ØØØØØØ1Ø Test N Flag
591 137A   26 2E              BNE   AFTSUB
592 137C   D7 89              STB   WORK1
593 137E   5F                 CLRB
594 137F   84 ØF              ANDA  #$ØF
595 1381   81 Ø9              CMPA  #9          Test b3 b2 b1 bØ
596 1383   23 Ø2              BLS   NSETH1
597 1385   CA 1Ø              ORB   #%ØØØ1ØØØØ Set H Flag
598            1387   NSETH1 EQU   *
599 1387   D7 8B              STB   F1
6ØØ 1389   D6 89              LDB   WORK1
6Ø1 138B   C5 1Ø              BITB  #%ØØØ1ØØØØ Test H Flag
6Ø2 138D   26 Ø3              BNE   SETH2
6Ø3 138F   1C DF              ANDCC #%11Ø11111 Clear H Flag (68Ø9)
6Ø4 1391      8E              FCB   $8E         LDX #  (for Skipping)
6Ø5 1392   1A 2Ø      SETH2   ORCC  #%ØØ1ØØØØØ Set H Flag (68Ø9)
6Ø6 1394   54                 LSRB              Get Carry (HalfCarry is not Move)
6Ø7 1395   96 8A      PAS2    LDA   A1
6Ø8 1397   19                 DAA
6Ø9 1398   97 8A              STA   A1          (Carry is not Move)
61Ø 139A   D6 8B              LDB   F1          (Carry is not Move)
611 139C   24 Ø2              BCC   NOCDAA
612 139E   CA Ø1              ORB   #%ØØØØØØØ1 Set Carry
613 13AØ   84 ØF      NOCDAA  ANDA  #$ØF
614 13A2   81 Ø9              CMPA  #$Ø9
615 13A4   23 2C              BLS   NOHDAA
616 13A6   C4 EF              ANDB  #%111Ø1111 Clear HalfCarry
617 13A8   2Ø 28              BRA   NOHDAA

619            13AA   AFTSUB EQU   *           DAA after Subtract
62Ø 13AA   ØF 89              CLR   WORK1
621 13AC   C5 1Ø              BITB  #%ØØØ1ØØØØ Test HalfCarry
622 13AE   27 Ø4              BEQ   HIGH
623            13BØ   LOWADJ EQU   *
624 13BØ   86 Ø6              LDA   #$Ø6
625 13B2   97 89              STA   WORK1
626            13B4   HIGH   EQU   *
627 13B4   C5 Ø1              BITB  #%ØØØØØØØ1 Test Carry
628 13B6   27 Ø6              BEQ   FLGDAA
629 13B8   86 6Ø              LDA   #$6Ø
63Ø 13BA   9A 89              ORA   WORK1
631 13BC   97 89              STA   WORK1
632 13BE   C4 Ø3      FLGDAA  ANDB  #%ØØØØØØ11 KEEP N and Carry
633 13CØ   ØØ 89              NEG   WORK1
634 13C2   96 8A              LDA   A1
635 13C4   9B 89              ADDA  WORK1
636 13C6   34 Ø1              PSHS  CC
637 13C8   97 8A              STA   A1
638 13CA   35 Ø2              PULS  A
639 13CC   85 2Ø              BITA  #%ØØ1ØØØØØ Test HalfCarry
64Ø 13CE   27 Ø2              BEQ   NOHDAA
641 13DØ   CA 1Ø              ORB   #%ØØØ1ØØØØ Set HalfCarry
642 13D2   96 8A      NOHDAA  LDA   A1
643 13D4   26 Ø3      NOTC3   BNE   NOZDAA
644 13D6   CA 4Ø              ORB   #%Ø1ØØØØØØ Set Zero
645 13D8   4D                 TSTA
646 13D9   2A Ø2      NOZDAA  BPL   NOSDAA
647 13DB   CA 8Ø              ORB   #%1ØØØØØØØ Set Sign
648            13DD   NOSDAA EQU   *
```


▶X1用の「A列車で行こう」がほしい。アートディンクさん早く作っておくれ。FDDもほしい。「猫とコンピュータ」が終わるのは悲しい。計算技術検定が目前なのにOh！MZのおかげで落ちそうだ。ハガキを書いている場合じゃない。　本間　晃（16）新潟県

```
650           13DD   PARI   EQU   *          Parity Check    A=Data B=Flag
651 13DD   D7 89            STB   WORK1      Save Flag
652 13DF   5F               CLRB             Clear Bit Counter
653 13E0   44        PARI1  LSRA
654 13E1   24 02            BCC   PARI2
655 13E3   5C               INCB
656 13E4   4D               TSTA
657 13E5   26 F9     PARI2  BNE   PARI1      If A=0 Then Check End
658 13E7   96 89            LDA   WORK1      Restore Flag
659 13E9   C5 01            BITB  #%00000001 Even or Odd ?
660 13EB   26 02            BNE   ODD1
661 13ED   8A 04            ORA   #%00000100 Set P/V
662 13EF   97 8B     ODD1   STA   F1         Store Flag
663 13F1   39               RTS

665           13F2   SLA    EQU   *
666 13F2   96 89            LDA   WORK1
667 13F4   48               ASLA
668 13F5   20 0F            BRA   FLGSFT

670           13F7   SRA    EQU   *
671 13F7   96 89            LDA   WORK1
672 13F9   47               ASRA
673 13FA   20 0A            BRA   FLGSFT

675           13FC   SLL    EQU   *
676 13FC   96 89            LDA   WORK1
677 13FE   48               ASLA
678 13FF   8A 01            ORA   #%00000001 (Carry is not move)
679 1401   20 03            BRA   FLGSFT

681           1403   SRL    EQU   *
682 1403   96 89            LDA   WORK1
683 1405   44               LSRA

685           1406   FLGSFT EQU   *
686 1406   1F A9            TFR   CC,B
687 1408   C4 01            ANDB  #%00000001 Get Carry
688 140A   D7 89            STB   WORK1      Save Flag
689 140C   D6 82            LDB   DATA3
690 140E   C1 06            CMPB  #$06
691 1410   26 03            BNE   STr2
692 1412   A7 C4            STA   ,U
693 1414      8E            FCB   $8E        LDX #  (For Skipping)
694           1415   STr2   EQU   *
695 1415   A7 A5            STA   B,Y
696           1417   FLGST2 EQU   *
697 1417   D6 89            LDB   WORK1      Restore Flag
698 1419   4D               TSTA
699 141A   20 B8            BRA   NOTC3

701           141C   SCF    EQU   *
702 141C   96 8B            LDA   F1
703 141E   84 C4            ANDA  #%11000100
704 1420   8A 01            ORA   #%00000001 Set CARRY
705 1422   97 8B            STA   F1
706 1424   39               RTS

708           1425   CPL    EQU   *
709 1425   03 8A            COM   A1         1's Complement Acc
710 1427   96 8B            LDA   F1         Get Flag
711 1429   8A 12            ORA   #%00010010 Set N and H
712 142B   97 8B            STA   F1         Store Flag
713 142D   39               RTS

715           142E   CCF    EQU   *
716 142E   96 8B            LDA   F1
717 1430   85 01            BITA  #%00000001
718 1432   26 07            BNE   RESCY
719 1434   84 C4            ANDA  #%11000100 Reset N
720 1436   8A 01            ORA   #%00000001
721 1438   97 8B            STA   F1
722 143A   39               RTS
723 143B   84 C4     RESCY  ANDA  #%11000100
724 143D   97 8B            STA   F1
725 143F   39               RTS

727                  *===================================

729           1440   COL1   EQU   *          $40~$7F
730 1440   96 81            LDA   DATA2
731 1442   D6 82            LDB   DATA3
732 1444   81 06            CMPA  #$06
733 1446   27 10            BEQ   LDmHLr

735 1448   C1 06            CMPB  #$06
736 144A   27 05            BEQ   LDrmHL
737                  *                   LD r,s
738 144C   E6 A5            LDB   B,Y        Get s
739 144E   E7 A6            STB   A,Y
740 1450   39               RTS

742           1451   LDrmHL EQU   *          LD r,(HL)
743 1451   9D AF            JSR   INDEX3
744 1453   E6 C4            LDB   ,U         Get (HL)
745 1455   E7 A6            STB   A,Y
746 1457   39               RTS

748           1458   LDmHLr EQU   *          LD (HL),r
749 1458   C1 06            CMPB  #$06
750 145A   27 08            BEQ   HALT       Ignore HALT
751 145C   1F 98            TFR   B,A
752 145E   9D AF            JSR   INDEX3
753 1460   A6 A6            LDA   A,Y
754 1462   A7 C4            STA   ,U
755 1464   39        HALT   RTS

757                  *==========================================

759           1465   COL2   EQU   *          $80~$BF
760 1465   96 82            LDA   DATA3
761 1467   81 06            CMPA  #$06
762 1469   26 05            BNE   GETr1
763 146B   9D 9D            JSR   INDEX2
764 146D   E6 C4            LDB   ,U
765 146F      8E            FCB   $8E        LDX #  (for Skipping)
766 1470   E6 A6     GETr1  LDB   A,Y
767 1472   D7 89     JPCOL2 STB   WORK1
768 1474   96 81            LDA   DATA2
769 1476   48               ASLA
770 1477   8E 147E          LDX   #JTBL2
771 147A   AE 86            LDX   A,X
772 147C   6E 84            JMP   ,X

774           147E   JTBL2  EQU   *
775 147E      1496          FDB   ADD
776 1480      14BE          FDB   ADC
777 1482      14C3          FDB   SUB
778 1484      14E5          FDB   SBC
779 1486      150E          FDB   AND
780 1488      1523          FDB   XOR
781 148A      152C          FDB   OR
782 148C      1535          FDB   CP

784           148E   COM3_6 EQU   *          ADD A,n|SUB n|AND n|OR n
785           148E   COM3_E EQU   *          ADC A,n|SBC A,n|XOR n|CP n
786 148E   9E 9B            LDX   PC
787 1490   E6 80            LDB   ,X+        Get n
788 1492   9F 9B            STX   PC
789 1494   20 DC            BRA   JPCOL2

791           1496   ADD    EQU   *
792 1496   1C FE            ANDCC #%11111110 Clear Carry (6809)
793                  *                   ADD A,r or (HL)
794 1498   D6 8A     ADD1   LDB   A1         (Carry is not move)
795 149A   D9 89            ADCB  WORK1
796 149C   1F A8            TFR   CC,A
797 149E   D7 8A            STB   A1
798 14A0   5F               CLRB             B=Flag
799 14A1   85 20            BITA  #%00100000 Test HalfCarry
800 14A3   27 02            BEQ   NOTH5
801 14A5   CA 10            ORB   #%00010000 Set HalfCarry
802 14A7   44        NOTH5  LSRA             Test Carry
803 14A8   24 02            BCC   NOTC5
804 14AA   CA 01            ORB   #%00000001 Set Carry
805 14AC   44        NOTC5  LSRA             Test OverFlow
806 14AD   24 02            BCC   NOTV5
807 14AF   CA 04            ORB   #%00000100 Set OverFlow
808 14B1   44        NOTV5  LSRA             Test Zero
809 14B2   24 02            BCC   NOTZ5
810 14B4   CA 40            ORB   #%01000000 Set Zero
811 14B6   44        NOTZ5  LSRA             Test Negative
812 14B7   24 02            BCC   NOTS5
813 14B9   CA 80            ORB   #%10000000 Set Sign
814 14BB   D7 8B     NOTS5  STB   F1         Store Flag
815 14BD   39               RTS

817           14BE   ADC    EQU   *
818 14BE   D6 8B            LDB   F1
819 14C0   54               LSRB             Get Carry
820 14C1   20 D5            BRA   ADD1

822           14C3   SUB    EQU   *
823 14C3   C6 02            LDB   #%00000010
824 14C5   D7 8B     SBC1   STB   F1
825 14C7   D6 8A            LDB   A1         Get Acc
826 14C9   D7 80            STB   DATA       Save Acc before Subtract
827 14CB   D0 89            SUBB  WORK1
828 14CD   1F A8            TFR   CC,A
829 14CF   D7 8A            STB   A1
830 14D1   D6 89     SUBCP  LDB   WORK1
831 14D3   C4 0F            ANDB  #$0F
832 14D5   D7 89            STB   WORK1
833 14D7   D6 80            LDB   DATA
834 14D9   C4 0F            ANDB  #$0F
835 14DB   D1 89            CMPB  WORK1      Test HalfCarry
836 14DD   D6 8B            LDB   F1         (Carry is not move)
837 14DF   24 C6            BCC   NOTH5
838 14E1   CA 10            ORB   #%00010000 Set HalfCarry
839 14E3   20 C2            BRA   NOTH5

841           14E5   SBC    EQU   *
842 14E5   D6 8B            LDB   F1
843 14E7   C5 01            BITB  #%00000001 Test CARRY
844 14E9   27 D8            BEQ   SUB
845 14EB   D6 8A            LDB   A1
846 14ED   C4 0F            ANDB  #$0F
847 14EF   27 03            BEQ   HLFSET
848 14F1   C6 02            LDB   #%00000010
849 14F3      8E            FCB   $8E        LDX #  (for Skipping)
850 14F4   C6 12     HLFSET LDB   #%00010010 Set HalfCarry
851 14F6   D7 8B            STB   F1
852 14F8   D6 8A            LDB   A1
853 14FA   C0 01            SUBB  #$01       Subtract Carry
854 14FC   1F A8            TFR   CC,A
855 14FE   D7 8A            STB   A1
856 1500   D6 8B            LDB   F1
857 1502   44               LSRA             Test Carry
858 1503   24 02            BCC   NCC1
859 1505   CA 01            ORB   #%00000001 Set Carry
860 1507   44        NCC1   LSRA             Test OverFlow
861 1508   24 BB            BCC   SBC1
862 150A   CA 04            ORB   #%00000100 Set OverFlow
863 150C   20 B7            BRA   SBC1

865           150E   AND    EQU   *          AND r or (HL)
866 150E   96 8A            LDA   A1
867 1510   94 89            ANDA  WORK1
868 1512   97 8A            STA   A1
869 1514   C6 10            LDB   #$10       B=Flag(H(flag)=1)
870 1516   4D        FLGLOG TSTA
871 1517   26 03            BNE   NOTZ7
872 1519   CA 40            ORB   #%01000000 Set Zero
873 151B   4D               TSTA
874 151C   2A 02     NOTZ7  BPL   NOTS7
875 151E   CA 80            ORB   #%10000000 Set Sign
876 1520   7E 13DD   NOTS7  JMP   PARI

878           1523   XOR    EQU   *          XOR r or (HL)
879 1523   96 8A            LDA   A1
880 1525   98 89            EORA  WORK1
881 1527   97 8A            STA   A1
882 1529   5F               CLRB             B=Flag
883 152A   20 EA            BRA   FLGLOG

885           152C   OR     EQU   *
886 152C   96 8A            LDA   A1
887 152E   9A 89            ORA   WORK1
888 1530   97 8A            STA   A1
889 1532   5F               CLRB
890 1533   20 E1            BRA   FLGLOG

892           1535   CP     EQU   *
893 1535   C6 02            LDB   #%00000010
894 1537   D7 8B            STB   F1
895 1539   D6 8A            LDB   A1
896 153B   D7 80            STB   DATA       Store Acc before Subtract
897 153D   D0 89            SUBB  WORK1
898 153F   1F A8            TFR   CC,A
899 1541   20 8E            BRA   SUBCP

901                  *=========================================================

903           1543   COL3   EQU   *
904 1543   96 80            LDA   DATA
905 1545   84 0F            ANDA  #$0F
906 1547   48               ASLA
907 1548   8E 154F          LDX   #JTBL3
908 154B   AE 86            LDX   A,X
909 154D   6E 84            JMP   ,X

911           154F   JTBL3  EQU   *
912 154F      156F          FDB   COM3_0
913 1551      15BE          FDB   COM3_1
914 1553      1615          FDB   COM3_2
915 1555      1670          FDB   COM3_3
916 1557      16A6          FDB   COM3_4
917 1559      1781          FDB   COM3_5
918 155B      148E          FDB   COM3_6
919 155D      1719          FDB   COM3_7
920 155F      156F          FDB   COM3_8
921 1561      15DC          FDB   COM3_9
922 1563      1615          FDB   COM3_A
923 1565      1670          FDB   COM3_B
924 1567      16A6          FDB   COM3_C
925 1569      179D          FDB   COM3_D
926 156B      148E          FDB   COM3_E
927 156D      1719          FDB   COM3_F

929                  *-----------------------------------------------------------

931           156F   COM3_0 EQU   *          RET cc
932           156F   COM3_8 EQU   *
933 156F   96 8B            LDA   F1
934 1571   D6 81            LDB   DATA2
935 1573   58               ASLB
936 1574   8E 157B          LDX   #JTB3_0
937 1577   AE 85            LDX   B,X
938 1579   6E 84            JMP   ,X

940           157B   JTB3_0 EQU   *
941 157B      1596          FDB   RETNZ
942 157D      159B          FDB   RETZ
943 157F      15A0          FDB   RETNC
```

▶うちの学校の図書館には，パソコン誌はOh！MZしかありません。
木下　達也（15）愛媛県

```
944 1581       15A5           FDB   RETC
945 1583       15AA           FDB   RETPO
946 1585       15AF           FDB   RETPE
947 1587       15B4           FDB   RETP
948 1589       15B9           FDB   RETM

950            158B   RET     EQU   *
951 158B   9E 99              LDX   SP
952 158D   EC 81              LDD   ,X++         Pop Return Address
953 158F   9F 99              STX   SP
954 1591   D7 9B              STB   PC
955 1593   97 9C              STA   PC+1
956 1595   39                 RTS

958            1596   RETNZ   EQU   *
959 1596   85 40              BITA  #%01000000
960 1598   27 F1              BEQ   RET
961 159A   39                 RTS

963            159B   RETZ    EQU   *
964 159B   85 40              BITA  #%01000000
965 159D   26 EC              BNE   RET
966 159F   39                 RTS

968            15A0   RETNC   EQU   *
969 15A0   85 01              BITA  #%00000001
970 15A2   27 E7              BEQ   RET
971 15A4   39                 RTS

973            15A5   RETC    EQU   *
974 15A5   85 01              BITA  #%00000001
975 15A7   26 E2              BNE   RET
976 15A9   39                 RTS

978            15AA   RETPO   EQU   *
979 15AA   85 04              BITA  #%00000100
980 15AC   27 DD              BEQ   RET
981 15AE   39                 RTS

983            15AF   RETPE   EQU   *
984 15AF   85 04              BITA  #%00000100
985 15B1   26 D8              BNE   RET
986 15B3   39                 RTS

988            15B4   RETP    EQU   *
989 15B4   85 80              BITA  #%10000000
990 15B6   27 D3              BEQ   RET
991 15B8   39                 RTS

993            15B9   RETM    EQU   *
994 15B9   85 80              BITA  #%10000000
995 15BB   26 CE              BNE   RET
996 15BD   39                 RTS

998                   *------------------------------------------------

1000           15BE   COM3_1  EQU   *
1001 15BE  96 81              LDA   DATA2
1002 15C0  81 06              CMPA  #$06
1003 15C2  27 0D              BEQ   POPAF
1004                  *                          POP rp
1005 15C4  33 A6              LEAU  A,Y
1006 15C6  9E 99              LDX   SP
1007 15C8  EC 81              LDD   ,X++
1008 15CA  9F 99              STX   SP
1009 15CC  E7 C4              STB   ,U
1010 15CE  A7 41              STA   1,U
1011 15D0  39                 RTS

1013           15D1   POPAF   EQU   *            POP AF
1014 15D1  9E 99              LDX   SP
1015 15D3  EC 81              LDD   ,X++
1016 15D5  9F 99              STX   SP
1017 15D7  D7 8A              STB   A1
1018 15D9  97 8B              STA   F1
1019 15DB  39                 RTS

1021                  *--------------------

1023           15DC   COM3_9  EQU   *
1024 15DC  96 81              LDA   DATA2
1025 15DE  84 06              ANDA  #%00000110
1026 15E0  8E 15E7            LDX   #JTB3_9
1027 15E3  AE 86              LDX   A,X
1028 15E5  6E 84              JMP   ,X

1030           15E7   JTB3_9  EQU   *
1031 15E7      158B           FDB   RET
1032 15E9      15EF           FDB   EXX
1033 15EB      160B           FDB   JPmHL
1034 15ED      1610           FDB   LDSPHL

1036           15EF   EXX     EQU   *
1037 15EF 108E 1083           LDY   #B1
1038 15F3  8E 108C            LDX   #B2
1039 15F6  C6 03              LDB   #3
1040 15F8  D7 89              STB   WORK1
1041 15FA  EC A4      EXXLOP  LDD   ,Y
1042 15FC  EE 84              LDU   ,X
1043 15FE  ED 81              STD   ,X++
1044 1600  EF A1              STU   ,Y++
1045 1602  0A 89              DEC   WORK1
1046 1604  26 F4              BNE   EXXLOP
1047 1606 108E 1083           LDY   #B1
1048 160A  39                 RTS

1050           160B   JPmHL   EQU   *
1051 160B  DC 87              LDD   H1
1052 160D  DD 9B              STD   PC
1053 160F  39                 RTS

1055           1610   LDSPHL  EQU   *
1056 1610  DC 87              LDD   H1
1057 1612  DD 99              STD   SP
1058 1614  39                 RTS

1060                  *------------------------------------------------

1062           1615   COM3_2  EQU   *
1063           1615   COM3_A  EQU   *
1064 1615  96 8B              LDA   F1
1065 1617  D6 81              LDB   DATA2
1066 1619  58                 ASLB
1067 161A  8E 1621            LDX   #JTB3_2
1068 161D  AE 85              LDX   B,X
1069 161F  6E 84              JMP   ,X

1071           1621   JTB3_2  EQU   *
1072 1621      163A           FDB   JPNZ
1073 1623      1640           FDB   JPZ
1074 1625      1646           FDB   JPNC
1075 1627      164C           FDB   JPC
1076 1629      1652           FDB   JPPO
1077 162B      1658           FDB   JPPE
1078 162D      165E           FDB   JPP
1079 162F      1664           FDB   JPM

1081           1631   JP      EQU   *
1082 1631  9E 9B              LDX   PC
1083 1633  E6 84              LDB   ,X
1084 1635  A6 01              LDA   1,X
1085 1637  DD 9B              STD   PC
1086 1639  39                 RTS

1088           163A   JPNZ    EQU   *
1089 163A  85 40              BITA  #%01000000
1090 163C  27 F3              BEQ   JP
1091 163E  20 28              BRA   SKIP2

1093           1640   JPZ     EQU   *
1094 1640  85 40              BITA  #%01000000
1095 1642  26 ED              BNE   JP
1096 1644  20 22              BRA   SKIP2

1098           1646   JPNC    EQU   *
1099 1646  85 01              BITA  #%00000001
1100 1648  27 E7              BEQ   JP
1101 164A  20 1C              BRA   SKIP2

1103           164C   JPC     EQU   *
1104 164C  85 01              BITA  #%00000001
1105 164E  26 E1              BNE   JP
1106 1650  20 16              BRA   SKIP2

1108           1652   JPPO    EQU   *
1109 1652  85 04              BITA  #%00000100
1110 1654  27 DB              BEQ   JP
1111 1656  20 10              BRA   SKIP2

1113           1658   JPPE    EQU   *
1114 1658  85 04              BITA  #%00000100
1115 165A  26 D5              BNE   JP
1116 165C  20 0A              BRA   SKIP2

1118           165E   JPP     EQU   *
1119 165E  85 80              BITA  #%10000000
1120 1660  27 CF              BEQ   JP
1121 1662  20 04              BRA   SKIP2

1123           1664   JPM     EQU   *
1124 1664  85 80              BITA  #%10000000
1125 1666  26 C9              BNE   JP

1127           1668   SKIP2   EQU   *
1128 1668  DC 9B              LDD   PC
1129 166A  C3 0002            ADDD  #2
1130 166D  DD 9B              STD   PC
1131 166F  39                 RTS

1133                  *------------------------------------------------

1135           1670   COM3_3  EQU   *
1136           1670   COM3_B  EQU   *
1137 1670  96 81              LDA   DATA2
1138 1672  48                 ASLA               A=A*2
1139 1673  8E 167A            LDX   #JTB3_3
1140 1676  AE 86              LDX   A,X
1141 1678  6E 84              JMP   ,X

1143           167A   JTB3_3  EQU   *
1144 167A      1631           FDB   JP
1145 167C      17B0           FDB   COM_CB
1146 167E      1188           FDB   INnA         Ignore IN n,A
1147 1680      1188           FDB   OUTAn        Ignore OUT A,n
1148 1682      168A           FDB   EmSPHL
1149 1684      169D           FDB   EXDEHL
1150 1686      115C           FDB   DI
1151 1688      115C           FDB   EI

1153           168A   EmSPHL  EQU   *
1154 168A  9E 99              LDX   SP
1155 168C  E6 84              LDB   ,X
1156 168E  A6 01              LDA   1,X
1157 1690  9E 87              LDX   H1
1158 1692  DD 87              STD   H1
1159 1694  1F 10              TFR   X,D
1160 1696  9E 99              LDX   SP
1161 1698  E7 84              STB   ,X
1162 169A  A7 01              STA   1,X
1163 169C  39                 RTS

1165           169D   EXDEHL  EQU   *
1166 169D  9E 87              LDX   H1
1167 169F  DC 85              LDD   D1
1168 16A1  DD 87              STD   H1
1169 16A3  9F 85              STX   D1
1170 16A5  39                 RTS

1172                  *------------------------------------------------

1174           16A6   COM3_4  EQU   *
1175           16A6   COM3_C  EQU   *
1176 16A6  96 81              LDA   DATA2
1177 16A8  48                 ASLA
1178 16A9  8E 16B0            LDX   #JTB3_4
1179 16AC  AE 86              LDX   A,X
1180 16AE  6E 84              JMP   ,X

1182           16B0   JTB3_4  EQU   *
1183 16B0      16D3           FDB   CALLNZ
1184 16B2      16DB           FDB   CALLZ
1185 16B4      16E3           FDB   CALLNC
1186 16B6      16EB           FDB   CALLC
1187 16B8      16F3           FDB   CALLPO
1188 16BA      16FB           FDB   CALLPE
1189 16BC      1703           FDB   CALLP
1190 16BE      170B           FDB   CALLM

1192           16C0   CALL    EQU   *            CALL nn
1193 16C0  9E 9B              LDX   PC
1194 16C2  E6 80              LDB   ,X+          Get nn
1195 16C4  A6 80              LDA   ,X+
1196 16C6  DD 9B              STD   PC
1197 16C8  DE 99      CALL1   LDU   SP
1198 16CA  1F 10              TFR   X,D
1199 16CC  A7 C2              STA   ,-U          Push Return Address
1200 16CE  E7 C2              STB   ,-U
1201 16D0  DF 99              STU   SP
1202 16D2  39                 RTS

1204           16D3   CALLNZ  EQU   *
1205 16D3  96 8B              LDA   F1
1206 16D5  85 40              BITA  #%01000000
1207 16D7  27 E7              BEQ   CALL
1208 16D9  20 36              BRA   SKIP22

1210           16DB   CALLZ   EQU   *
1211 16DB  96 8B              LDA   F1
1212 16DD  85 40              BITA  #%01000000
1213 16DF  26 DF              BNE   CALL
1214 16E1  20 2E              BRA   SKIP22

1216           16E3   CALLNC  EQU   *
1217 16E3  96 8B              LDA   F1
1218 16E5  85 01              BITA  #%00000001
1219 16E7  27 D7              BEQ   CALL
1220 16E9  20 26              BRA   SKIP22

1222           16EB   CALLC   EQU   *
1223 16EB  96 8B              LDA   F1
1224 16ED  85 01              BITA  #%00000001
1225 16EF  26 CF              BNE   CALL
1226 16F1  20 1E              BRA   SKIP22

1228           16F3   CALLPO  EQU   *
1229 16F3  96 8B              LDA   F1
1230 16F5  85 04              BITA  #%00000100
1231 16F7  27 C7              BEQ   CALL
1232 16F9  20 16              BRA   SKIP22

1234           16FB   CALLPE  EQU   *
1235 16FB  96 8B              LDA   F1
1236 16FD  85 04              BITA  #%00000100
1237 16FF  26 BF              BNE   CALL
```


▶７月号のグラフィック特集，よかったです。私の場合，タイルチェンジよりも，白黒処理のあと反転させ，そして黒の縦線によるブラインド加工でハードコピー，というパターンを多用しています。
木下　研一（16）佐賀県

```
1238 1701    20 0E            BRA   SKIP22

1240              1703  CALLP  EQU   *
1241 1703    96 8B            LDA   F1
1242 1705    85 80            BITA  #%10000000
1243 1707    27 B7            BEQ   CALL
1244 1709    20 06            BRA   SKIP22

1246              170B  CALLM  EQU   *
1247 170B    96 8B            LDA   F1
1248 170D    85 80            BITA  #%10000000
1249 170F    26 AF            BNE   CALL

1251              1711  SKIP22 EQU   *
1252 1711    DC 9B            LDD   PC
1253 1713    C3 0002          ADDD  #2
1254 1716    DD 9B            STD   PC
1255 1718    39               RTS

1257                    *---------------------------------------------

1259              1719  COM3_7 EQU   *
1260              1719  COM3_F EQU   *             RESTART
1261 1719    96 81            LDA   DATA2
1262 171B    81 01            CMPA  #$01
1263 171D    27 14            BEQ   RST08H        Return to 6809
1264 171F    8E 172B          LDX   #RSTTBL
1265 1722    E6 86            LDB   A,X
1266 1724    4F               CLRA
1267 1725    9E 9B            LDX   PC
1268 1727    DD 9B            STD   PC
1269 1729    20 9D            BRA   CALL1

1271              172B  RSTTBL EQU   *
1272 172B       00            FCB   $00
1273 172C       08            FCB   $08
1274 172D       10            FCB   $10
1275 172E       18            FCB   $18
1276 172F       20            FCB   $20
1277 1730       28            FCB   $28
1278 1731       30            FCB   $30
1279 1732       38            FCB   $38

1281              1733  RST08H EQU   *
1282 1733    35 10            PULS  X             Drop Return Address
1283 1735    9E 9B            LDX   PC
1284 1737    A6 80            LDA   ,X+
1285 1739    9F 9B            STX   PC
1286 173B    DE 99            LDU   SP
1287 173D    B7 0082          STA   >FUNCNO
1288 1740    1F 10            TFR   X,D
1289 1742    A7 C2            STA   ,-U           PUSH PC
1290 1744    E7 C2            STB   ,-U
1291 1746    8E 1099          LDX   #SP
1292 1749    C6 02            LDB   #2            PUSH IY,IX
1293 174B    8D 27            BSR   RSTSUB
1294 174D    8E 1089          LDX   #WORK1
1295 1750    C6 03            LDB   #3            PUSH HL,DE,BC
1296 1752    8D 20            BSR   RSTSUB
1297 1754    D6 8A            LDB   A1
1298 1756    96 8B            LDA   F1
1299 1758    ED C3            STD   ,--U          PUSH AF
1300 175A    8E 1092          LDX   #INDEX        .
1301 175D    C6 03            LDB   #3            PUSH HL',DE',BC'
1302 175F    8D 13            BSR   RSTSUB
1303 1761    D6 93            LDB   A2
1304 1763    96 94            LDA   F2
1305 1765    ED C3            STD   ,--U          PUSH AF'
1306 1767    ED C3            STD   ,--U
1307 1769    DF 99            STU   SP
1308 176B    35 08            PULS  DP
1309                 00       SETDP $00
1310 176D    1F 30            TFR   U,D
1311 176F    D7 80            STB   ZSTKPT        Store Stack Pointer
1312 1771    97 81            STA   ZSTKPT+1
1313 1773    39               RTS                 Return to 6809

1315                 10       SETDP $10

1317              1774  RSTSUB EQU   *
1318 1774    D7 89            STB   WORK1
1319 1776    A6 82      RSTLOP LDA   ,-X
1320 1778    E6 82            LDB   ,-X
1321 177A    ED C3            STD   ,--U
1322 177C    0A 89            DEC   WORK1
1323 177E    26 F6            BNE   RSTLOP
1324 1780    39               RTS

1326                    *-------------------------------------------

1328              1781  COM3_5 EQU   *
1329 1781    96 81            LDA   DATA2
1330 1783    81 06            CMPA  #$06
1331 1785    27 0B            BEQ   PUSHAF
1332                                              PUSH rp
1333 1787    EC A6            LDD   A,Y
1334 1789    9E 99            LDX   SP
1335 178B    A7 82            STA   ,-X
1336 178D    E7 82            STB   ,-X
1337 178F    9F 99            STX   SP
1338 1791    39               RTS

1340              1792  PUSHAF EQU   *
1341 1792    D6 8A            LDB   A1
1342 1794    96 8B            LDA   F1
1343 1796    9E 99            LDX   SP
1344 1798    ED 83            STD   ,--X
1345 179A    9F 99            STX   SP
1346 179C    39               RTS

1348                    *-------------------------

1350              179D  COM3_D EQU   *
1351 179D    96 81            LDA   DATA2
1352 179F    84 06            ANDA  #%00000110
1353 17A1    8E 17A8          LDX   #JTB3_D
1354 17A4    AE 86            LDX   A,X
1355 17A6    6E 84            JMP   ,X

1357              17A8  JTB3_D EQU   *
1358 17A8       16C0          FDB   CALL
1359 17AA       183B          FDB   COM_DD
1360 17AC       1899          FDB   COM_ED
1361 17AE       1856          FDB   COM_FD

1363                    *=====================================================

1365              17B0  COM_CB EQU   *
1366 17B0    9D CB            JSR   GETDTA
1367 17B2    34 04            PSHS  B
1368 17B4    96 82            LDA   DATA3
1369 17B6    81 06            CMPA  #$06
1370 17B8    26 05            BNE   GETr2
1371 17BA    9D 9D            JSR   INDEX2
1372 17BC    E6 C4            LDB   ,U
1373 17BE       8E            FCB   $8E           LDX #  (for Skipping)
1374 17BF    E6 A6      GETr2  LDB   A,Y
1375 17C1    D7 89      JPCMCB STB   WORK1
1376 17C3    35 02            PULS  A
1377 17C5    8E 17CC          LDX   #JCM_CB
1378 17C8    AE 86            LDX   A,X
1379 17CA    6E 84            JMP   ,X

1381              17CC  JCM_CB EQU   *
1382 17CC       17D4          FDB   SHIFT
1383 17CE       17EE          FDB   BIT
1384 17D0       1816          FDB   RES
1385 17D2       180B          FDB   SETBIT

1387              17D4  SHIFT  EQU   *
1388 17D4    96 81            LDA   DATA2
1389 17D6    48               ASLA                A=A*2
1390 17D7    8E 17DE          LDX   #JSHIFT
1391 17DA    AE 86            LDX   A,X
1392 17DC    6E 84            JMP   ,X

1394              17DE  JSHIFT EQU   *
1395 17DE       1307          FDB   RLC
1396 17E0       1329          FDB   RRC
1397 17E2       133D          FDB   RL
1398 17E4       1358          FDB   RR
1399 17E6       13F2          FDB   SLA
1400 17E8       13F7          FDB   SRA
1401 17EA       13FC          FDB   SLL
1402 17EC       1403          FDB   SRL

1404              17EE  BIT    EQU   *
1405 17EE    96 89            LDA   WORK1
1406 17F0    D6 81            LDB   DATA2
1407 17F2    8E 182B          LDX   #BITMSK
1408 17F5    A5 85            BITA  B,X
1409 17F7    26 09            BNE   FLGBT1
1410 17F9    D6 8B            LDB   F1
1411 17FB    C4 01            ANDB  #%00000001 Keep Carry
1412 17FD    CA 50            ORB   #%01010000 Set HalfCarry and Zero
1413 17FF    D7 8B            STB   F1
1414 1801    39               RTS
1415 1802    D6 8B      FLGBT1 LDB   F1
1416 1804    C4 01            ANDB  #%00000001
1417 1806    CA 10            ORB   #%00010000 Set HalfCarry
1418 1808    D7 8B            STB   F1
1419 180A    39               RTS

1421              180B  SETBIT EQU   *
1422 180B    96 89            LDA   WORK1
1423 180D    D6 81            LDB   DATA2
1424 180F    8E 182B          LDX   #BITMSK
1425 1812    AA 85            ORA   B,X
1426 1814    20 09            BRA   STORE2

1428              1816  RES    EQU   *
1429 1816    96 89            LDA   WORK1
1430 1818    D6 81            LDB   DATA2
1431 181A    8E 1833          LDX   #RESMSK
1432 181D    A4 85            ANDA  B,X

1434              181F  STORE2 EQU   *
1435 181F    D6 82            LDB   DATA3
1436 1821    C1 06            CMPB  #$06
1437 1823    26 03            BNE   STr3
1438 1825    A7 C4            STA   ,U
1439 1827    39               RTS
1440              1828  STr3   EQU   *
1441 1828    A7 A5            STA   B,Y
1442 182A    39               RTS

1444              182B  BITMSK EQU   *
1445 182B       01            FCB   $01           BIT 0
1446 182C       02            FCB   $02           .
1447 182D       04            FCB   $04           .
1448 182E       08            FCB   $08           .
1449 182F       10            FCB   $10           .
1450 1830       20            FCB   $20           .
1451 1831       40            FCB   $40           .
1452 1832       80            FCB   $80           BIT 7

1454              1833  RESMSK EQU   *
1455 1833       FE            FCB   $FE           BIT 0
1456 1834       FD            FCB   $FD           .
1457 1835       FB            FCB   $FB           .
1458 1836       F7            FCB   $F7           .
1459 1837       EF            FCB   $EF           .
1460 1838       DF            FCB   $DF           .
1461 1839       BF            FCB   $BF           .
1462 183A       7F            FCB   $7F           BIT 7

1464                    *========================================================

1466              183B  COM_DD EQU   *             for IX
1467 183B    0C 92            INC   INDEX
1468 183D    9D CB            JSR   GETDTA
1469 183F    C1 02            CMPB  #2            $40~$7F
1470 1841    27 4C            BEQ   NOT_CB
1471 1843    9E 95            LDX   IX
1472 1845    DE 87            LDU   H1
1473 1847    9F 87            STX   H1
1474 1849    DF 95            STU   IX
1475 184B    8D 24            BSR   DD_FD
1476 184D    9E 95            LDX   IX
1477 184F    DE 87            LDU   H1
1478 1851    9F 87            STX   H1
1479 1853    DF 95            STU   IX
1480 1855    39               RTS

1482              1856  COM_FD EQU   *             for IY
1483 1856    03 92            COM   INDEX
1484 1858    9D CB            JSR   GETDTA
1485 185A    C1 02            CMPB  #2            $40~$7F
1486 185C    27 31            BEQ   NOT_CB
1487 185E    9E 97            LDX   IY
1488 1860    DE 87            LDU   H1
1489 1862    9F 87            STX   H1
1490 1864    DF 97            STU   IY
1491 1866    8D 09            BSR   DD_FD
1492 1868    9E 97            LDX   IY
1493 186A    DE 87            LDU   H1
1494 186C    9F 87            STX   H1
1495 186E    DF 97            STU   IY
1496 1870    39               RTS

1498              1871  DD_FD  EQU   *
1499 1871    96 80            LDA   DATA
1500 1873    81 CB            CMPA  #$CB
1501 1875    26 18            BNE   NOT_CB
1502 1877    9E 9B            LDX   PC
1503 1879    EC 84            LDD   ,X
1504 187B    E7 84            STB   ,X
1505 187D    A7 01            STA   1,X
1506 187F    34 10            PSHS  X
1507 1881    BD 17B0          JSR   COM_CB
1508 1884    35 10            PULS  X
1509 1886    EC 84            LDD   ,X
1510 1888    E7 84            STB   ,X
1511 188A    A7 01            STA   1,X
1512 188C    0F 92            CLR   INDEX
1513 188E    39               RTS

1515              188F  NOT_CB EQU   *
1516 188F    8E 110E          LDX   #JPTBL
1517 1892    AE 85            LDX   B,X
1518 1894    AD 84            JSR   ,X
1519 1896    0F 92            CLR   INDEX
1520 1898    39               RTS

1522                    *========================================================

1524              1899  COM_ED EQU   *
1525 1899    9D CB            JSR   GETDTA
1526 189B    C1 04            CMPB  #$04
1527 189D 1027 00E9           LBEQ  REPEAT
1528 18A1    96 82            LDA   DATA3
1529 18A3    48               ASLA
1530 18A4    8E 18AB          LDX   #JTB_ED
1531 18A7    AE 86            LDX   A,X
```

▶ブラインドタッチ上達のアイテム，それは「アロンアルファ」を L と S に１滴ずつかけて３分待てばでき上がり。
四田　誠治（22）愛媛県

```
1532 18A9   6E 84           JMP    ,X

1534             18AB  JTB_ED EQU    *
1535 18AB        115C         FDB    PORT       Ignore IN A,(C)
1536 18AD        115C         FDB    PORT       Ignore OUT A,(C)
1537 18AF        18BB         FDB    SBCADC
1538 18B1        1952         FDB    LDmnrp
1539 18B3        1941         FDB    NEG
1540 18B5        158B         FDB    RET        Only Return
1541 18B7        115C         FDB    IM012      Ignore IM0,IM1 and IM2
1542 18B9        1A9B         FDB    RRDRLD

1544             18BB  SBCADC EQU    *
1545 18BB   D6 81             LDB    DATA2
1546 18BD   C5 01             BITB   #%00000001
1547 18BF   1F A8             TFR    CC,A
1548 18C1   C4 06             ANDB   #%00000110
1549 18C3   C1 06             CMPB   #$06
1550 18C5   26 03             BNE    GETss1
1551 18C7   DE 99             LDU    SP
1552 18C9      8E             FCB    $8E        LDX #  (for Skipping)
1553 18CA   EE A5      GETss1 LDU    B,Y
1554 18CC   DF 80             STU    DATA
1555 18CE   1F 8A      SKPSAD TFR    A,CC
1556 18D0   27 42             BEQ    SBCss

1558             18D2  ADCss  EQU    *          ADC HL,ss
1559 18D2   D6 8B             LDB    F1
1560 18D4   54                LSRB              Test Carry
1561 18D5   24 17             BCC    ADDss
1562 18D7   DC 87             LDD    H1
1563 18D9   C3 0001           ADDD   #1         Add Carry
1564 18DC   34 01             PSHS   CC
1565 18DE   DD 87             STD    H1
1566 18E0   35 02             PULS   A
1567 18E2   5F                CLRB
1568 18E3   44                LSRA              Test Carry
1569 18E4   24 02             BCC    NOCADC
1570 18E6   CA 01             ORB    #%00000001 Set Carry
1571 18E8   44         NOCADC LSRA              Test OverFlow
1572 18E9   24 04             BCC    ADDss2
1573 18EB   CA 04             ORB    #%00000100 Set OverFlow
1574 18ED      21             FCB    $21        BRN  (for Skipping)

1576 18EE   5F         ADDss  CLRB
1577 18EF   D7 8B      ADDss2 STB    F1
1578 18F1   DC 87             LDD    H1         Get ss
1579 18F3   D3 80             ADDD   DATA
1580 18F5   34 01             PSHS   CC
1581 18F7   DD 87             STD    H1         Store HL
1582 18F9   35 02             PULS   A
1583 18FB   D6 8B      FLGSUB LDB    F1
1584 18FD   44         FLGSBC LSRA              Test Carry
1585 18FE   24 02             BCC    NOTC8
1586 1900   CA 01             ORB    #%00000001 Set Carry
1587 1902   44         NOTC8  LSRA              Test OverFlow
1588 1903   24 02             BCC    NOTV8
1589 1905   CA 04             ORB    #%00000100 Set P/V
1590 1907   44         NOTV8  LSRA              Test Zero
1591 1908   24 02             BCC    NOTZ8
1592 190A   CA 40             ORB    #%01000000 Set Zero
1593 190C   44         NOTZ8  LSRA              Test Negative
1594 190D   24 02             BCC    NOTS8
1595 190F   CA 80             ORB    #%10000000 Set Sign
1596 1911   D7 8B      NOTS8  STB    F1
1597 1913   39                RTS

1599             1914  SBCss  EQU    *          SBC HL,ss
1600 1914   D6 8B             LDB    F1
1601 1916   54                LSRB              Test Carry
1602 1917   24 18             BCC    SUBss
1603 1919   DC 87             LDD    H1
1604 191B   83 0001           SUBD   #1
1605 191E   34 01             PSHS   CC
1606 1920   DD 87             STD    H1
1607 1922   35 02             PULS   A
1608 1924   C6 02             LDB    #%00000010
1609 1926   44                LSRA              Test Carry
1610 1927   24 02             BCC    NOCSBC
1611 1929   CA 01             ORB    #%00000001
1612 192B   44         NOCSBC LSRA              Test OverFlow
1613 192C   24 05             BCC    SUBss2
1614 192E   CA 04             ORB    #%00000100 Set P/V
1615 1930      8E             FCB    $8E        LDX #  (for Skipping)

1617 1931   C6 02      SUBss  LDB    #%00000010 Set N
1618 1933   D7 8B      SUBss2 STB    F1
1619 1935   DC 87             LDD    H1         Get HL
1620 1937   93 80             SUBD   DATA       Sub ss
1621 1939   34 01             PSHS   CC
1622 193B   DD 87             STD    H1
1623 193D   35 02             PULS   A
1624 193F   20 BA             BRA    FLGSUB

1626             1941  NEG    EQU    *
1627 1941   D6 8A             LDB    A1
1628 1943   00 8A             NEG    A1
1629 1945   1F A8             TFR    CC,A
1630 1947   C4 0F             ANDB   #$0F
1631 1949   26 03             BNE    SETHLF
1632 194B   5F                CLRB
1633 194C   20 AF             BRA    FLGSBC
1634 194E   C6 20      SETHLF LDB    #%00100000 Set HalfCarry
1635 1950   20 AB             BRA    FLGSBC

1637             1952  LDmnrp EQU    *
1638 1952   D6 81             LDB    DATA2
1639 1954   C5 01             BITB   #%00000001
1640 1956   34 01             PSHS   CC
1641 1958   C4 06             ANDB   #%00000110
1642 195A   C1 06             CMPB   #$06
1643 195C   26 04             BNE    GETss2
1644 195E   CE 1099           LDU    #SP
1645 1961      8E             FCB    $8E        LDX #  (for Skipping)
1646 1962   33 A5      GETss2 LEAU   B,Y
1647 1964   35 01      SKPLDm PULS   CC
1648 1966   26 11             BNE    LDddmn

1650             1968  LDmndd EQU    *          LD dd,(nn)
1651 1968   9E 9B             LDX    PC
1652 196A   E6 80             LDB    ,X+        Get nn
1653 196C   A6 80             LDA    ,X+
1654 196E   9F 9B             STX    PC
1655 1970   1F 01             TFR    D,X
1656 1972   EC C4             LDD    ,U         Get dd
1657 1974   E7 84             STB    ,X         Store to (nn)
1658 1976   A7 01             STA    1,X
1659 1978   39                RTS

1661             1979  LDddmn EQU    *          LD (nn),dd
1662 1979   9E 9B             LDX    PC
1663 197B   E6 80             LDB    ,X+        Get nn
1664 197D   A6 80             LDA    ,X+
1665 197F   9F 9B             STX    PC
1666 1981   1F 01             TFR    D,X
1667 1983   EC 84             LDD    ,X         Get (nn)
1668 1985   E7 C4             STB    ,U         Store to dd
1669 1987   A7 41             STA    1,U
1670 1989   39                RTS

1672             198A  REPEAT EQU    *
1673 198A   D6 82             LDB    DATA3
1674 198C   C5 02             BITB   #%00000010
1675 198E   27 01             BEQ    MEM
1676 1990   39                RTS               Port  (INIR, OTDR and so on)

1678             1991  MEM    EQU    *          Memory
1679 1991   C5 01             BITB   #%00000001
1680 1993   27 14             BEQ    LD?

1682             1995  CP?    EQU    *
1683 1995   D6 81             LDB    DATA2
1684 1997   C4 03             ANDB   #%00000011
1685 1999   58                ASLB
1686 199A   8E 19A1           LDX    #JTBCP?
1687 199D   AE 85             LDX    B,X
1688 199F   6E 84             JMP    ,X

1690             19A1  JTBCP? EQU    *
1691 19A1        19BD         FDB    CPI
1692 19A3        19C7         FDB    CPD
1693 19A5        1A08         FDB    CPIR
1694 19A7        1A15         FDB    CPDR

1696             19A9  LD?    EQU    *
1697 19A9   D6 81             LDB    DATA2
1698 19AB   C4 03             ANDB   #%00000011
1699 19AD   58                ASLB
1700 19AE   8E 19B5           LDX    #JTBLD?
1701 19B1   AE 85             LDX    B,X
1702 19B3   6E 84             JMP    ,X

1704             19B5  JTBLD? EQU    *
1705 19B5        1A22         FDB    LDI
1706 19B7        1A30         FDB    LDD
1707 19B9        1A51         FDB    LDIR
1708 19BB        1A72         FDB    LDDR

1710             19BD  CPI    EQU    *
1711 19BD   9E 87             LDX    H1         Get HL
1712 19BF   E6 80             LDB    ,X+        Get (HL)
1713 19C1   D7 89             STB    WORK1
1714 19C3   9F 87             STX    H1
1715 19C5   20 0A             BRA    CP??

1717             19C7  CPD    EQU    *
1718 19C7   9E 87             LDX    H1         Get HL
1719 19C9   E6 84             LDB    ,X         Get (HL)
1720 19CB   D7 89             STB    WORK1
1721 19CD   30 1F             LEAX   -1,X       HL=HL-1
1722 19CF   9F 87             STX    H1

1724 19D1   96 8A      CP??   LDA    A1         Get Acc
1725 19D3   97 80             STA    DATA       Save Acc before Subtract
1726 19D5   91 89             CMPA   WORK1      A-(HL)
1727 19D7   1F A8             TFR    CC,A       Save Flag
1728 19D9   D6 89             LDB    WORK1
1729 19DB   C4 0F             ANDB   #$0F
1730 19DD   D7 89             STB    WORK1
1731 19DF   D6 80             LDB    DATA
1732 19E1   C4 0F             ANDB   #$0F
1733 19E3   D1 89             CMPB   WORK1
1734 19E5   D6 8B             LDB    F1         (Carry is not move)
1735 19E7   C4 01             ANDB   #%00000001 Keep Carry (Carry is not move)
1736 19E9   CA 02             ORB    #%00000010 Set N (Carry is not move)
1737 19EB   24 02             BCC    NOTH9
1738 19ED   CA 10             ORB    #%00010000 Set Half
1739 19EF   85 08      NOTH9  BITA   #%00001000 Test Negative
1740 19F1   27 02             BEQ    NOTS9
1741 19F3   CA 80             ORB    #%10000000 Set Sign
1742 19F5   85 04      NOTS9  BITA   #%00000100 Test Zero
1743 19F7   27 02             BEQ    NOTZ9
1744 19F9   CA 40             ORB    #%01000000
1745 19FB   9E 83      NOTZ9  LDX    B1
1746 19FD   30 1F             LEAX   -1,X
1747 19FF   9F 83             STX    B1
1748 1A01   27 02             BEQ    NOTV9
1749 1A03   CA 04             ORB    #%00000100 Set V
1750 1A05   D7 8B      NOTV9  STB    F1
1751 1A07   39                RTS

1753             1A08  CPIR   EQU    *
1754 1A08   8D B3             BSR    CPI
1755 1A0A   D6 8B             LDB    F1
1756 1A0C   C5 40             BITB   #%01000000 Test Zero
1757 1A0E   26 04             BNE    RETCP      Return If A=(HL)
1758 1A10   C5 04             BITB   #%00000100 Test P/V
1759 1A12   26 F4             BNE    CPIR       Repeat If BC<>0
1760 1A14   39         RETCP  RTS

1762             1A15  CPDR   EQU    *
1763 1A15   8D B0             BSR    CPD
1764 1A17   D6 8B             LDB    F1
1765 1A19   C5 40             BITB   #%01000000 Test Zero
1766 1A1B   26 F7             BNE    RETCP      Return If A=(HL)
1767 1A1D   C5 04             BITB   #%00000100 Test P/V
1768 1A1F   26 F4             BNE    CPDR       Repeat If BC<>0
1769 1A21   39                RTS

1771             1A22  LDI    EQU    *
1772 1A22   9E 85             LDX    D1         Get Destination
1773 1A24   DE 87             LDU    H1         Get Source
1774 1A26   A6 C0             LDA    ,U+
1775 1A28   A7 80             STA    ,X+
1776 1A2A   9F 85             STX    D1
1777 1A2C   DF 87             STU    H1
1778 1A2E   20 10             BRA    LD??

1780             1A30  LDD    EQU    *
1781 1A30   9E 85             LDX    D1
1782 1A32   DE 87             LDU    H1
1783 1A34   A6 C4             LDA    ,U
1784 1A36   A7 84             STA    ,X
1785 1A38   30 1F             LEAX   -1,X
1786 1A3A   33 5F             LEAU   -1,U
1787 1A3C   9F 85             STX    D1
1788 1A3E   DF 87             STU    H1
1789 1A40   D6 8B      LD??   LDB    F1
1790 1A42   C4 C1             ANDB   #%11000001
1791 1A44   9E 83             LDX    B1         Get Byte Count
1792 1A46   30 1F             LEAX   -1,X
1793 1A48   9F 83             STX    B1
1794 1A4A   27 02             BEQ    KEEPPV
1795 1A4C   CA 04             ORB    #%00000100 Set P/V
1796 1A4E   D7 8B      KEEPPV STB    F1
1797 1A50   39                RTS

1799             1A51  LDIR   EQU    *
1800 1A51   9E 83             LDX    B1         Counter
1801 1A53   DE 85             LDU    D1         Destination
1802 1A55 109E 87             LDY    H1         Source
1803 1A58   A6 A0      LDILOP LDA    ,Y+
1804 1A5A   A7 C0             STA    ,U+
1805 1A5C   30 1F             LEAX   -1,X
1806 1A5E   26 F8             BNE    LDILOP
1807 1A60   9F 83             STX    B1
1808 1A62   DF 85             STU    D1
1809 1A64 109F 87             STY    H1
1810 1A67 108E 1083           LDY    #REGTOP
1811 1A6B   96 8B             LDA    F1
1812 1A6D   84 C1             ANDA   #%11000001
1813 1A6F   97 8B             STA    F1
1814 1A71   39                RTS

1816             1A72  LDDR   EQU    *
1817 1A72   9E 83             LDX    B1         Counter
1818 1A74   DE 85             LDU    D1         Destination
1819 1A76 109E 87             LDY    H1         Source
1820 1A79   31 21             LEAY   1,Y
1821 1A7B   33 41             LEAU   1,U
1822 1A7D   A6 A2      LDDLOP LDA    ,-Y
1823 1A7F   A7 C2             STA    ,-U
1824 1A81   30 1F             LEAX   -1,X
1825 1A83   26 F8             BNE    LDDLOP
```


▶ X68000, X1, MZ-2500もいいけれど, MZ-1500にだって1000文字PCGというものがある。使いこなせるやつは果たして何人？　　照井　清和（16）秋田県

```
1826 1A85    31 3F             LEAY  -1,Y
1827 1A87    33 5F             LEAU  -1,U
1828 1A89    9F 83             STX   B1
1829 1A8B    DF 85             STU   D1
1830 1A8D  109F 87             STY   H1
1831 1A90  108E 1083           LDY   #REGTOP
1832 1A94    96 8B             LDA   F1
1833 1A96    84 C1             ANDA  #%11000001
1834 1A98    97 8B             STA   F1
1835 1A9A    39         LDAIR  RTS

1837              1A9B  RRDRLD EQU   *
1838 1A9B    D6 81             LDB   DATA2
1839 1A9D    C5 04             BITB  #%100
1840 1A9F    27 F9             BEQ   LDAIR       Ignore LD A,I,LD A,R and so on
1841 1AA1    C5 02             BITB  #%010
1842 1AA3    26 F5             BNE   LDAIR       ???
1843 1AA5    C5 01             BITB  #%001
1844 1AA7    27 2F             BEQ   RRD

1846              1AA9  RLD    EQU   *
1847 1AA9    9E 87             LDX   H1          Get HL
1848 1AAB    96 8A             LDA   A1
1849 1AAD    1F 89             TFR   A,B
1850 1AAF    84 F0             ANDA  #$F0        Keep b7~b4
1851 1AB1    97 89             STA   WORK1
1852 1AB3    68 84             ASL   ,X
1853 1AB5    59                ROLB
1854 1AB6    68 84             ASL   ,X
1855 1AB8    59                ROLB
1856 1AB9    68 84             ASL   ,X
1857 1ABB    59                ROLB
1858 1ABC    68 84             ASL   ,X
1859 1ABE    59                ROLB
1860 1ABF    1F 98             TFR   B,A
1861 1AC1    84 0F             ANDA  #$0F
1862 1AC3    9A 89             ORA   WORK1
1863 1AC5    97 8A             STA   A1
1864 1AC7    C4 F0             ANDB  #$F0
1865 1AC9    54                LSRB
1866 1ACA    54                LSRB
1867 1ACB    54                LSRB
1868 1ACC    54                LSRB
1869 1ACD    EA 84             ORB   ,X
1870 1ACF    E7 84             STB   ,X
1871 1AD1    D6 8B             LDB   F1
1872 1AD3    C4 01             ANDB  #%00000001 Keep Carry
1873 1AD5    7E 1516           JMP   FLGLOG

1875              1AD8  RRD    EQU   *
1876 1AD8    9E 87             LDX   H1
1877 1ADA    96 8A             LDA   A1
1878 1ADC    1F 89             TFR   A,B
1879 1ADE    C4 F0             ANDB  #$F0        Keep b7~b4
1880 1AE0    D7 89             STB   WORK1
1881 1AE2    44                LSRA
1882 1AE3    66 84             ROR   ,X
1883 1AE5    46                RORA
1884 1AE6    66 84             ROR   ,X
1885 1AE8    46                RORA
1886 1AE9    66 84             ROR   ,X
1887 1AEB    46                RORA
1888 1AEC    66 84             ROR   ,X
1889 1AEE    46                RORA
1890 1AEF    44                LSRA
1891 1AF0    44                LSRA
1892 1AF1    44                LSRA
1893 1AF2    44                LSRA
1894 1AF3    9A 89             ORA   WORK1
1895 1AF5    97 8A             STA   A1
1896 1AF7    D6 8B             LDB   F1
1897 1AF9    C4 01             ANDB  #%00000001
1898 1AFB    7E 1516           JMP   FLGLOG

1900              1AFD  EMUEND EQU   *-1
```

リスト13　エミュレータ用変更点ソースリスト

```
 1          0893   IRQBRK EQU   $0893
 2          10E9   Z80    EQU   $10E9
 3          10F1   PC0000 EQU   $10F1
 4          1015   CMDHDL EQU   $1015
 5          2100   HOT    EQU   $2100

 7                 *=========================

 9 0878                   ORG   $0878

11 0878   16 07BD         LBRA  CMDENT
12                 *---
13 0886                   ORG   $0886

15 0886   7E 103D         JMP   IRQPAT
16                 *---
17 08B8                   ORG   $08B8

19 08B8   39              RTS
20                 *---
21 1033                   ORG   $1033

23 1033   BD 10E9         JSR   Z80        Sleep 6809 & Return Z80
24 1036   20 DD           BRA   CMDHDL

26 1038   BD 10F1  CMDENT JSR   PC0000
27 103B   20 D8           BRA   CMDHDL

29           103D  IRQPAT EQU   *
30 103D   34 0A           PSHS  DP,A
31 103F   4F              CLRA
32 1040   1F 8B           TFR   A,DP
33 1042   17 F84E         LBSR  IRQBRK
34 1045   35 0A           PULS  DP,A
35 1047   3B              RTI

37                 *---
38 1B1F                   ORG   $1B1F

40 1B1F      C3           FCB   $C3         ' JP HOT
41 1B20      00           FCB   <HOT,>HOT
42                 *---
43 1E4F                   ORG   $1E4F

45 1E4F      C3           FCB   $C3         ' JP HOT
46 1E50      00           FCB   <HOT,>HOT
47                 *---
48 1F8E                   ORG   $1F8E

50 1F8E      C3           FCB   $C3         ' JP HOT
51 1F8F      00           FCB   <HOT,>HOT
```

リスト14　メモリエディタソースリスト

```
0000                 1 ;  MACHINE CODE INPUT TOOL
0000                 2 ;
0000                 3          OFFSET  3000H
3000                 4          ORG     3000H
3000                 5          ;
3000                 6 #HLHEX   EQU     1FB2H
3000                 7 #PRTHL   EQU     1FBEH
3000                 8 #PRTHX   EQU     1FC1H
3000                 9 #GETL    EQU     1FD3H
3000                10 #MSX     EQU     1FE5H
3000                11 #LETNL   EQU     1FEEH
3000                12 #PRNTS   EQU     1FF1H
3000                13 #PRINT   EQU     1FF4H
3000                14 #CSRSET  EQU     201EH
3000                15 #FLGET   EQU     2021H
3000                16          ;
3000                17 #KBFAD   EQU     1F76H
3000                18
3000                19 START:
3000 11 30 32       20          LD      DE,OPNING
3003 CD E5 1F       21          CALL    #MSX
3006 ED 5B 76 1F    22          LD      DE,(#KBFAD)
300A CD D3 1F       23          CALL    #GETL
300D 1A             24          LD      A,(DE)
300E FE 1B          25          CP      1BH
3010 C8             26          RET     Z
3011                27          ;
3011 CD B2 1F       28          CALL    #HLHEX
3014 38 EA          29          JR      C,START
3016 22 7B 32       30          LD      (ADRS),HL
3019 11 43 32       31          LD      DE,TITLE
301C CD E5 1F       32          CALL    #MSX
301F CD 27 30       33          CALL    DUMP
3022 CD D4 30       34          CALL    KEYIN
3025 18 D9          35          JR      START
3027                36 ;
3027                37 DUMP:
3027 21 00 04       38          LD      HL,0400H
302A CD 1E 20       39          CALL    #CSRSET
302D 2A 7B 32       40          LD      HL,(ADRS)
3030 0E 10          41          LD      C,16
3032 CD 3D 30       42 DUMP1:   CALL    LNDMP
3035 CD EE 1F       43          CALL    #LETNL
3038 0D             44          DEC     C
3039 20 F7          45          JR      NZ,DUMP1
303B 18 24          46          JR      COLSUM
303D                47 ;
303D                48 LNDMP:
303D E5             49          PUSH    HL
303E CD BE 1F       50          CALL    #PRTHL
3041 E1             51          POP     HL
3042 06 08          52          LD      B,8
3044 1E 00          53          LD      E,0
3046 CD F1 1F       54 LNDMP1:  CALL    #PRNTS
3049 7E             55          LD      A,(HL)
304A 23             56          INC     HL
304B 57             57          LD      D,A
304C CD C1 1F       58          CALL    #PRTHX
304F                59          ;
304F 7A             60          LD      A,D
3050 83             61          ADD     A,E
3051 5F             62          LD      E,A
3052                63          ;
3052 10 F2          64          DJNZ    LNDMP1
3054 CD F1 1F       65          CALL    #PRNTS
3057 3E 3A          66          LD      A,':'
3059 CD F4 1F       67          CALL    #PRINT
305C 7B             68          LD      A,E
305D CD C1 1F       69          CALL    #PRTHX
3060 C9             70          RET
3061                71
3061                72 COLSUM:
3061 21 00 14       73          LD      HL,1400H
3064 CD 1E 20       74          CALL    #CSRSET
3067 06 20          75          LD      B,32
3069 3E 2D          76          LD      A,'-'
306B CD F4 1F       77 CLSM1:   CALL    #PRINT
306E 10 FB          78          DJNZ    CLSM1
3070 CD EE 1F       79          CALL    #LETNL
3073                80          ;
3073 11 75 32       81          LD      DE,MES
3076 CD E5 1F       82          CALL    #MSX
3079                83          ;
3079 2A 7B 32       84          LD      HL,(ADRS)
307C 0E 08          85          LD      C,8
307E                86          ;
307E 06 10          87 CLSM3:   LD      B,16
3080 11 08 00       88          LD      DE,8
3083 AF             89          XOR     A
3084 86             90 CLSM4:   ADD     A,(HL)
3085 19             91          ADD     HL,DE
3086 10 FC          92          DJNZ    CLSM4
3088                93          ;
3088 CD C1 1F       94          CALL    #PRTHX
308B CD F1 1F       95          CALL    #PRNTS
308E 11 7F 00       96          LD      DE,127
3091 B7             97          OR      A
3092 ED 52          98          SBC     HL,DE
3094 0D             99          DEC     C
3095 20 E7         100          JR      NZ,CLSM3
3097               101          ;
3097 3E 3A         102          LD      A,':'
3099 C9            103          RET
309A               104          ;
309A CD F4 1F      105          CALL    #PRINT
309D 2A 7B 32      106          LD      HL,(ADRS)
30A0 06 80         107          LD      B,128
30A2               108 CRC:
30A2 56            109          LD      D,(HL)
30A3 5A            110          LD      E,D       ; dummy
30A4 23            111          INC     HL
30A5 05            112          DEC     B
```

▶吉田幸一さんの「Kくん」シリーズが面白い。　　山本　伸明（15）北海道

```
30A6 28 27          113             JR      Z,DUMP5
30A8                114             ;
30A8 5E             115             LD      E,(HL)
30A9 23             116             INC     HL
30AA 05             117             DEC     B
30AB 28 22          118             JR      Z,DUMP5 ; case of bytes < 3
30AD                119             ;
30AD D5             120 CRC1:       PUSH    DE      ; first 16 bits
30AE 1E 80          121             LD      E,80H   ; mask pattern
30B0 D9             122             EXX
30B1 E1             123             POP     HL      ; first 16 bits
30B2 D9             124             EXX
30B3                125             ;
30B3 7E             126 CRC2:       LD      A,(HL)  ; load 1 byte
30B4 A3             127             AND     E       ; get bit
30B5 28 01          128             JR      Z,CRC3  ; CY=0
30B7 37             129             SCF             ; CY=1
30B8                130             ;
30B8 D9             131 CRC3:       EXX
30B9 ED 6A          132             ADC     HL,HL   ; add the bit to HL
30BB 30 08          133             JR      NC,CRC4 ; non 16th bit
30BD                134             ;
30BD 3E 10          135             LD      A,10H
30BF AC             136             XOR     H
30C0 67             137             LD      H,A
30C1 3E 21          138             LD      A,21H   ; 1021H=X^12+X^5+1
30C3 AD             139             XOR     L
30C4 6F             140             LD      L,A     ; HL=HL XOR 1021H
30C5                141             ;
30C5 D9             142 CRC4:       EXX
30C6 CB 0B          143             RRC     E       ; rotate mask pattern
30C8                144             ;               ;     to get next bit
30C8 30 E9          145             JR      NC,CRC2 ; case of loop =< 8
30CA 23             146             INC     HL      ; next byte
30CB 10 E6          147             DJNZ    CRC2
30CD D9             148             EXX
30CE EB             149             EX      DE,HL
30CF                150 ;
30CF EB             151 DUMP5:      EX      DE,HL
30D0 CD BE 1F       152             CALL    #PRTHL
30D3 C9             153             RET
30D4                154
30D4                155 KEYIN:
30D4 21 05 04       156             LD      HL,0405H
30D7 22 7D 32       157             LD      (LOC),HL
30DA                158             ;
30DA 2A 7D 32       159 KEYIN1:     LD      HL,(LOC)
30DD CD 1E 20       160             CALL    #CSRSET
30E0 CD 21 20       161             CALL    #FLGET
30E3 01 19 00       162             LD      BC,25
30E6 21 CA 31       163             LD      HL,KYTBL
30E9 ED B1          164             CPIR
30EB 20 ED          165             JR      NZ,KEYIN1
30ED                166             ;
30ED FE 30          167             CP      '0'
30EF 30 34          168             JR      NC,UDKEY
30F1                169             ;
30F1 FE 1C          170             CP      1CH
30F3 38 26          171             JR      C,SPCKY
30F5                172 ;
30F5 20 05          173 CSRR:       JR      NZ,CSRL
30F7                174             ;
30F7 CD 71 31       175             CALL    CSRRGT
30FA 18 DE          176             JR      KEYIN1
30FC                177 ;
30FC FE 1D          178 CSRL:       CP      1DH
30FE 20 05          179             JR      NZ,CSRU
3100                180             ;
3100 CD 94 31       181             CALL    CSRLFT
3103 18 D5          182             JR      KEYIN1
3105                183 ;
3105 FE 1E          184 CSRU:       CP      1EH
3107 20 09          185             JR      NZ,CSRD
3109                186             ;
3109 06 10          187             LD      B,16
310B CD 94 31       188 CSRU1:      CALL    CSRLFT
310E 10 FB          189             DJNZ    CSRU1
3110 18 C8          190             JR      KEYIN1
3112                191 ;
3112 06 10          192 CSRD:       LD      B,16
3114 CD 71 31       193 CSRD1:      CALL    CSRRGT
3117 10 FB          194             DJNZ    CSRD1
3119 18 BF          195             JR      KEYIN1
311B                196 ;
311B 21 7D 32       197 SPCKY:      LD      HL,LOC
311E FE 0D          198             CP      0DH
3120 C0             199             RET     NZ
3121 36 05          200             LD      (HL),5
3123 18 B5          201             JR      KEYIN1
3125                202 ;
3125 FE 54          203 UDKEY:      CP      'T'
3127 20 0E          204             JR      NZ,UDKEY1
3129                205             ;
3129 2A 7B 32       206             LD      HL,(ADRS)
312C 11 80 00       207             LD      DE,128
312F B7             208             OR      A
3130 ED 52          209             SBC     HL,DE
3132 22 7B 32       210             LD      (ADRS),HL
3135 18 0E          211             JR      UDKEY2
3137                212             ;
3137 FE 47          213 UDKEY1:     CP      'G'
3139 20 0F          214             JR      NZ,RSTORE
313B                215             ;
313B 2A 7B 32       216             LD      HL,(ADRS)
313E 11 80 00       217             LD      DE,128
3141 19             218             ADD     HL,DE
3142 22 7B 32       219             LD      (ADRS),HL
3145                220             ;
3145 CD 27 30       221 UDKEY2:     CALL    DUMP
3148 18 8A          222             JR      KEYIN
314A                223 ;
314A FE 52          224 RSTORE:     CP      'R'
314C 20 05          225             JR      NZ,EDIT
314E                226             ;
314E CD 61 30       227             CALL    COLSUM
3151 18 81          228             JR      KEYIN
3153                229 ;
3153 CD E3 31       230 EDIT:       CALL    EDITION
3156 2A 7D 32       231             LD      HL,(LOC)
3159 2E 00          232             LD      L,0
315B CD 1E 20       233             CALL    #CSRSET
315E                234             ;
315E CD 27 32       235             CALL    YOFST
3161 5F             236             LD      E,A
3162 16 00          237             LD      D,0
3164 2A 7B 32       238             LD      HL,(ADRS)
3167 19             239             ADD     HL,DE
3168                240             ;
3168 CD 3D 30       241             CALL    LNDMP
316B CD 71 31       242             CALL    CSRRGT
316E C3 DA 30       243             JP      KEYIN1
3171                244
3171                245 CSRRGT:
3171 C5             246             PUSH    BC
3172 21 7D 32       247             LD      HL,LOC
3175 3E 1B          248             LD      A,27
3177 BE             249             CP      (HL)
3178 20 05          250             JR      NZ,CSRRT1
317A                251             ;
317A CD 89 31       252             CALL    NXLN
317D 18 08          253             JR      CSRRT2
317F 7E             254 CSRRT1:     LD      A,(HL)
3180 34             255             INC     (HL)
3181 CD B7 31       256             CALL    LOCHK
3184 20 01          257             JR      NZ,CSRRT2
3186 34             258             INC     (HL)
3187 C1             259 CSRRT2:     POP     BC
3188 C9             260             RET
3189                261 ;
3189 36 05          262 NXLN:       LD      (HL),5
318B 23             263             INC     HL
318C 7E             264             LD      A,(HL)
318D 34             265             INC     (HL)
318E FE 13          266             CP      19
3190 C0             267             RET     NZ
3191                268             ;
3191 36 04          269             LD      (HL),4
3193 C9             270             RET
3194                271
3194                272 CSRLFT:
3194 C5             273             PUSH    BC
3195 21 7D 32       274             LD      HL,LOC
3198 3E 05          275             LD      A,5
319A BE             276             CP      (HL)
319B 20 05          277             JR      NZ,CSLFT1
319D                278             ;
319D CD AC 31       279             CALL    BFRLN
31A0 18 08          280             JR      CSLFT2
31A2 7E             281 CSLFT1:     LD      A,(HL)
31A3 35             282             DEC     (HL)
31A4 CD B7 31       283             CALL    LOCHK
31A7 28 01          284             JR      Z,CSLFT2
31A9 35             285             DEC     (HL)
31AA C1             286 CSLFT2:     POP     BC
31AB C9             287             RET
31AC                288 ;
31AC 36 1B          289 BFRLN:      LD      (HL),27
31AE 23             290             INC     HL
31AF 7E             291             LD      A,(HL)
31B0 35             292             DEC     (HL)
31B1 FE 04          293             CP      4
31B3 C0             294             RET     NZ
31B4                295             ;
31B4 36 13          296             LD      (HL),19
31B6 C9             297             RET
31B7                298
31B7                299 LOCHK:
31B7 E5             300             PUSH    HL
31B8 01 08 00       301             LD      BC,8
31BB 21 C2 31       302             LD      HL,MOTN
31BE ED B1          303             CPIR
31C0 E1             304             POP     HL
31C1 C9             305             RET
31C2                306 ;
31C2 06 09 0C 0F    307 MOTN:       DEFB    6,9,12,15
31C6 12 15 18 1B    308             DEFB    18,21,24,27
31CA                309
31CA 0D 1B          310 KYTBL:      DEFB    0DH,1BH
31CC 1C 1D 1E 1F    311             DEFB    1CH,1DH,1EH,1FH
31D0 30 31 32 33    312             DEFB    '0','1','2','3'
31D4 34 35 36 37    313             DEFB    '4','5','6','7'
31D8 38 39 41 42    314             DEFB    '8','9','A','B'
31DC 43 44 45 46    315             DEFB    'C','D','E','F'
31E0 54 47 52       316             DEFB    'T','G','R'
31E3                317
31E3                318 EDITION:
31E3 CD 1F 32       319             CALL    TOHEX
31E6 47             320             LD      B,A
31E7 CD 27 32       321             CALL    YOFST
31EA 4F             322             LD      C,A
31EB                323             ;
31EB C5             324             PUSH    BC
31EC 3A 7D 32       325             LD      A,(LOC)
31EF D6 05          326             SUB     5
31F1 01 03 00       327             LD      BC,3
31F4 91             328 EDTION1:SUB     C
31F5 38 03          329             JR      C,EDTIN11
31F7 04             330             INC     B
31F8 18 FA          331             JR      EDTION1
31FA                332             ;
31FA 78             333 EDTIN11:LD      A,B
31FB C1             334             POP     BC
31FC 81             335             ADD     A,C
31FD                336             ;
31FD 5F             337             LD      E,A
31FE 16 00          338             LD      D,0
3200 2A 7B 32       339             LD      HL,(ADRS)
3203 19             340             ADD     HL,DE
3204                341             ;
3204 3A 7D 32       342             LD      A,(LOC)
3207 C5             343             PUSH    BC
3208 CD B7 31       344             CALL    LOCHK
320B C1             345             POP     BC
320C 7E             346             LD      A,(HL)
320D 20 05          347             JR      NZ,EDTION2
320F                348             ;
320F E6 F0          349             AND     0F0H
3211 B0             350             OR      B
3212 18 09          351             JR      EDTION3
3214 E6 0F          352 EDTION2:AND     00FH
3216 4F             353             LD      C,A
3217 78             354             LD      A,B
3218 87             355             ADD     A,A
3219 87             356             ADD     A,A
321A 87             357             ADD     A,A
321B 87             358             ADD     A,A
321C B1             359             OR      C
321D 77             360 EDTION3:LD      (HL),A
321E C9             361             RET
321F                362 ;
321F D6 30          363 TOHEX:      SUB     '0'
3221 FE 0A          364             CP      10
3223 D8             365             RET     C
3224                366             ;
3224 D6 07          367             SUB     7
3226 C9             368             RET
3227                369 ;
3227                370 YOFST:
3227 3A 7E 32       371             LD      A,(LOC+1)
322A D6 04          372             SUB     4
322C 87             373             ADD     A,A
322D 87             374             ADD     A,A
322E 87             375             ADD     A,A
322F C9             376             RET
3230                377
3230 0C             378 OPNING:     DEFB    0CH
3231 49 4E 50 55    379             DEFM    "INPUT START ADRS"
3235 54 20 53 54
3239 41 52 54 20
323D 41 44 52 53
3241 0D 00          380             DEFB    0DH,0
3243                381
3243 0C             382 TITLE:      DEFB    0CH
3244 44 55 4D 50    383             DEFM    "DUMP & EDIT"
3248 20 26 20 45
324C 44 49 54
324F 0D 0D          384             DEFB    0DH,0DH
3251 41 44 52 53    385             DEFM    "ADRS +0 +1 +2 +3 +4 +5 +6 +7 :SUM"
3255 20 2B 30 20
3259 2B 31 20 2B
325D 32 20 2B 33
3261 20 2B 34 20
3265 2B 35 20 2B
3269 36 20 2B 37
326D 20 3A 53 55
3271 4D
3272 0D 0D          386             DEFB    0DH,0DH
3274 00             387             DEFB    0
3275                388
3275 53 55 4D 3A    389 MES:        DEFM    "SUM: "
3279 20
327A 00             390             DEFB    0
327B                391
327B 00 90          392 ADRS:       DEFW    9000H
327D 00 00          393 LOC:        DEFW    0
OBJECT CODE END 627E
```

▶ちょっと聞いてもらえます？「太陽の神殿」ってあんなところで終わってしまうんですか？　やっと受験がすんでまめまめしくゲームを続け，いいとこまできたなあ，と思ったとたん，唐突にエンディングが流れてきて……。でもプレイ中にいきなり終わったこともあるからなあ。信じられないよぉ。
桂　尚子（18）青森県

リスト15 IPLソースリスト

```
  0                     *
  1                     *       Initial Program Loader for S-OS SWORD
  2                     *       Version 1.1  1986/05/25
  3                     *       Designed and coded by S.Kigoshi
  4                     *
  5 0300                        ORG    $0300
  6             FD              SETDP $FD
  7                     *
  8                     *       Symbols
  9                     *
 10             FC7F    BOTSTK EQU    $FC7F       Boot stack
 11             FC80    SHARED EQU    $FC80       Common memory

 13             FD03    BUZZER EQU    $FD03       Buzzer control
 14             FD05    SBSIN  EQU    $FD05       Sub-system status
 15             FD05    SBSOUT EQU    $FD05       Sub-system control
 16             FD0F    MEMODE EQU    $FD0F       Memory mode

 18             FE08    DREADS EQU    $FE08       5" floppy disk 1 sector read.

 20             0000    RQNO   EQU    0           Request number
 21             0001    RCBSTA EQU    1           I/O status
 22             0002    RCBDBA EQU    2           Data buffer addr.
 23             0004    RCBTRK EQU    4           Track
 24             0005    RCBSCT EQU    5           Sector
 25             0006    RCBSID EQU    6           Side
 26             0007    RCBUNT EQU    7           Unit

 28             0028    SMSIZE EQU    40          System module size

 30             0000    EOS    EQU    $00         End of string
 31             0007    BEL    EQU    $07         Buzzer
 32             000D    CR     EQU    $0D         Carriage return
 33             0011    SF     EQU    $11         Set field
 34                     *
 35                     *       Entry
 36                     *
 37 0300     16 0092    ENTRY  LBRA   IPL
 38                     *
 39                     *       Request control block
 40                     *
 41 0303        0A      RCB    FCB    $0A         5" floppy disk 1 sector read
 42 0304        00             FCB    0           Dummy
 43 0305        0800           FDB    $0800       Address = $0800
 44 0307        01             FCB    1           Track   = 1
 45 0308        01             FCB    1           Sector  = 1
 46 0309        00             FCB    0           Side    = 0
 47 030A        00             FCB    0           Unit    = 0
 48                     *
 49                     *       Message
 50                     *
 51 030B        07      MIPLST FCB    BEL
 52 030C        2A             FCC    '**** S-OS SWORD IPL started !'
 53 0329        0D             FCB    CR,EOS

 55 032B        07      MSMLCP FCB    BEL
 56 032C        2A             FCC    '**** System module loading completed !'
 57 0352        0D             FCB    CR,EOS

 59 0354        07      MBERR  FCB    BEL,SF,2
 60 0357        2A             FCC    '**** I/O fault detected !'
 61 0370        00             FCB    EOS
 62                     *
 63                     *       Sub system halt sub-routine
 64                     *
 65 0371     96 05      SUBHLT LDA    SBSIN
 66 0373     2B FC             BMI    SUBHLT
 67 0375     86 80             LDA    #%10000000
 68 0377     97 05             STA    SBSOUT
 69 0379     96 05      SUBWAI LDA    SBSIN
 70 037B     2A FC             BPL    SUBWAI
 71 037D     39                RTS
 72                     *
 73                     *       Display message sub-routine
 74                     *
 75 037E     8D F1      DSPMSG BSR    SUBHLT
 76 0380     CE FC82           LDU    #SHARED+2
 77 0383     CC 0300           LDD    #$0300
 78 0386     A7 C1             STA    ,U++         !
 79 0388     5C         DSPMS1 INCB
 80 0389     A6 80             LDA    ,X+
 81 038B     A7 C0             STA    ,U+
 82 038D     26 F9             BNE    DSPMS1
 83 038F     F7 FC83           STB    SHARED+3
 84 0392     0F 05      DSPMS2 CLR    SBSOUT
 85 0394     39                RTS
 86                     *
 87                     *       Initial program loader main program
 88                     *
 89 0395     1A 50      IPL    ORCC   #%01010000
 90 0397   10CE FC7F           LDS    #BOTSTK
 91 039B     86 FD             LDA    #$FD
 92 039D     1F 8B             TFR    A,DP
 93 039F     97 0F             STA    MEMODE
 94 03A1     86 01             LDA    #1
 95 03A3     97 03             STA    BUZZER
 96 03A5     8D CA             BSR    SUBHLT
 97 03A7     CC 0C3F           LDD    #$0C00+%00111111
 98 03AA     FD FC82           STD    SHARED+2
 99 03AD     8D E3             BSR    DSPMS2
100 03AF     30 8DFF58         LEAX   MIPLST,PCR
101 03B3     8D C9             BSR    DSPMSG
102 03B5     30 8DFF4A         LEAX   RCB,PCR
103 03B9     EE 02             LDU    RCBDBA,X
104 03BB     C6 28             LDB    #SMSIZE
105 03BD     34 54      IPLOOP PSHS   U,X,B
106 03BF     BD FE08           JSR    DREADS
107 03C2     35 54             PULS   U,X,B
108 03C4     4D                TSTA
109 03C5     26 2C             BNE    IPLERR
110 03C7     5A                DECB
111 03C8     27 20             BEQ    IPLEXT
112 03CA     33 C90100         LEAU   $100,U
113 03CE     EF 02             STU    RCBDBA,X
114 03D0     6C 05             INC    RCBSCT,X
115 03D2     A6 05             LDA    RCBSCT,X
116 03D4     81 10             CMPA   #16
117 03D6     23 E5             BLS    IPLOOP
118 03D8     86 01             LDA    #1
119 03DA     A7 05             STA    RCBSCT,X
120 03DC     6C 06             INC    RCBSID,X
121 03DE     A6 06             LDA    RCBSID,X
122 03E0     81 01             CMPA   #1
123 03E2     23 D9             BLS    IPLOOP
124 03E4     6F 06             CLR    RCBSID,X
125 03E6     6C 04             INC    RCBTRK,X
126 03E8     20 D3             BRA    IPLOOP

128 03EA     30 8DFF3D IPLEXT  LEAX   MSMLCP,PCR
129 03EE     8D 8E             BSR    DSPMSG
130 03F0     7E 0800           JMP    $0800
131                     *
132                     *       Error trap
133                     *
134 03F3     30 8DFF5D IPLERR  LEAX   MBERR,PCR
135 03F7     8D 85             BSR    DSPMSG
136 03F9     20 FE             BRA    *
137                     *
138                     *       End of initial program loader
139                     *
```

リスト16 システムジェネレータソースリスト

```
  0 2000                        ORG    $2000
  1                     *
  2                     *       Symbol equates
  3                     *
  4             FBFA    BIOS   EQU    $FBFA

  6             0000    RQNO   EQU    0
  7             0001    RCBSTA EQU    1
  8             0002    RCBDBA EQU    2
  9             0004    RCBTRK EQU    4
 10             0005    RCBSCT EQU    5
 11             0006    RCBSID EQU    6
 12             0007    RCBUNT EQU    7
 13                     *
 14                     *       System genelate program entry
 15                     *
 16 2000     20 68      ENTRY  BRA    SYSGEN
 17                     *
 18                     *       Request control block
 19                     *
 20 2002        09      RCB    FCB    $09         'DWRITE'
 21 2003        0001           RMB    1           I/O status
 22 2004        0002           RMB    2           Data buffer addr.
 23 2006        0001           RMB    1           Track
 24 2007        0001           RMB    1           Sector
 25 2008        0001           RMB    1           Side
 26 2009        00             FCB    0           Unit
 27                     *
 28                     *       Disk map
 29                     *
 30 200A        2100    DSKMAP FDB    $2100       Addr.      IPL
 31 200C        00             FCB    0           Track
 32 200D        01             FCB    1           Sector
 33 200E        00             FCB    0           Side
 34 200F        01             FCB    1           Number of sector

 36 2010        2200           FDB    $2200       FAT
 37 2012        00             FCB    0
 38 2013        0F             FCB    15
 39 2014        00             FCB    0
 40 2015        02             FCB    2

 42 2016        2400           FDB    $2400       Directory
 43 2018        00             FCB    0
 44 2019        01             FCB    1
 45 201A        01             FCB    1
 46 201B        10             FCB    16

 48 201C        3800           FDB    $3800       System program
 49 201E        01             FCB    1
 50 201F        01             FCB    1
 51 2020        00             FCB    0
 52 2021        28             FCB    40          or 43 (with Memory Editor)

 54 2022        0000           FDB    0           Dummy
 55 2024        00             FCB    0
 56 2025        00             FCB    0
 57 2026        00             FCB    0
 58 2027        00             FCB    0
 59                     *
 60                     *       Write data to disk
 61                     *
 62 2028     34 40      WRTDSK PSHS   U
 63 202A     30 8CD5           LEAX   RCB,PCR
 64 202D     EC C4             LDD    ,U
 65 202F     ED 02             STD    2,X
 66 2031     EC 42             LDD    2,U
 67 2033     ED 04             STD    4,X
 68 2035     EC 44             LDD    4,U
 69 2037     A7 06             STA    6,X
 70 2039     EE 02             LDU    2,X
 71 203B     AD 9FFBFA WRTLOP  JSR    [BIOS]
 72 203F     25 26             BCS    WRTERR
 73 2041     5A                DECB
 74 2042     27 20             BEQ    WRTEXT
 75 2044     33 C90100         LEAU   $100,U
 76 2048     EF 02             STU    RCBDBA,X
 77 204A     6C 05             INC    RCBSCT,X
 78 204C     A6 05             LDA    RCBSCT,X
 79 204E     81 10             CMPA   #16
 80 2050     23 E9             BLS    WRTLOP
 81 2052     86 01             LDA    #1
 82 2054     A7 05             STA    RCBSCT,X
 83 2056     6C 06             INC    RCBSID,X
 84 2058     A6 06             LDA    RCBSID,X
 85 205A     81 01             CMPA   #1
 86 205C     23 DD             BLS    WRTLOP
 87 205E     6F 06             CLR    RCBSID,X
 88 2060     6C 04             INC    RCBTRK,X
 89 2062     20 D7             BRA    WRTLOP

 91 2064     5F         WRTEXT CLRB
 92 2065     35 C0             PULS   PC,U

 94 2067     53         WRTERR COMB
 95 2068     35 C0             PULS   PC,U
 96                     *
 97                     *       System genelater main routine
 98                     *
 99 206A     33 8C9D    SYSGEN LEAU   DSKMAP,PCR
100 206D     A6 45      SYSLOP LDA    5,U
101 206F     27 08             BEQ    SYSEND
102 2071     8D B5             BSR    WRTDSK
103 2073     25 04             BCS    SYSEND
104 2075     33 46             LEAU   6,U
105 2077     20 F4             BRA    SYSLOP

107 2079     39         SYSEND RTS
108                     *
109                     *       End of 'SYSGEN'
110                     *
```

▶今月，やっと即戦力の文章をユーカラに移すことができました。いままで両方のソフトに同じテキストを苦労して入力していたのが嘘のようです。ファイルコンバータ，これはじつに素晴しいです。

安藤　紀之 (42) 北海道

リスト17 FORMAT & SYSGEN ソースリスト

```
0000              1 ;+++++++++++++++++++++++++++++++++++++
0000              2 ;  Logical Format & System Generate
0000              3 ;
0000              4 ;                for FM-7
0000              5 ;+++++++++++++++++++++++++++++++++++++
0000              6
7000              7          ORG     7000H
7000              8
7000              9 #LTNL    EQU     1FEEH
7000             10 #MPRNT   EQU     1FE2H
7000             11 #PRINT   EQU     1FF4H
7000             12 #FLGET   EQU     2021H
7000             13 #DRDSB   EQU     2000H
7000             14 #DWTSB   EQU     2003H
7000             15 #ERROR   EQU     2033H
7000             16 #DSK     EQU     1F5DH
7000             17 #FATBF   EQU     $1F62
7000             18 #MXTRK   EQU     1F66H
7000             19 #FATPOS  EQU     1F5EH
7000             20 #DIRPS   EQU     1F60H
7000             21 #BELL    EQU     1FC4H
7000             22
7000 AF          23 RETRY    XOR     A
7001 32 6B 73    24          LD      (LOAD),A
7004 CD E2 1F    25 AGAIN0   CALL    #MPRNT
7007 0C          26          DEFB    0CH
7008 31 29 20    27          DEFM    "1) Logical Format"
700B 4C 6F 67
700E 69 63 61
7011 6C 20 46
7014 6F 72 6D
7017 61 74
7019 0D          28          DEFB    0DH
701A 32 29 20    29          DEFM    "2) & System Generate"
701D 26 20 53
7020 79 73 74
7023 65 6D 20
7026 47 65 6E
7029 65 72 61
702C 74 65
702E 0D          30          DEFB    0DH
702F 33 29 20    31          DEFM    "3) End of Job"
7032 45 6E 64
7035 20 6F 66
7038 20 4A 6F
703B 62
703C 0D 0D       32          DEFB    0DH:0DH
703E 49 6E 70    33          DEFM    "Input Number "
7041 75 74 20
7044 4E 75 6D
7047 62 65 72
704A 20
704B 00          34          DEFB    0
704C CD 21 20    35 KEYIN1   CALL    #FLGET
704F FE 1B       36          CP      1BH
7051 C8          37          RET     Z
7052 FE 33       38          CP      '3'
7054 20 06       39          JR      NZ,FMAT
7056 3E 0C       40 ENDFMT   LD      A,0CH
7058 CD F4 1F    41          CALL    #PRINT
705B C9          42          RET
705C             43          ;
705C FE 31       44 FMAT     CP      '1'
705E 38 EC       45          JR      C,KEYIN1
7060 FE 33       46          CP      '2'+1
7062 30 E8       47          JR      NC,KEYIN1
7064 32 6C 73    48          LD      (WORKNO),A
7067 CD F4 1F    49          CALL    #PRINT
706A FE 31       50          CP      '1'
706C CA 0F 72    51          JP      Z,FORMAT1
706F             52          ;
706F 3A 6B 73    53          LD      A,(LOAD)
7072 B7          54          OR      A
7073 C2 02 71    55          JP      NZ,SYSWT1        ;If System Loa
7076             56
7076             57 ;        LOAD SYSTEM
7076             58
7076 CD E2 1F    59 SYSRD1   CALL    #MPRNT
7079 0D 0D       60          DEFB    0DH:0DH
707B 53 6F 75    61          DEFM    "Source Drive Name = "
707E 72 63 65
7081 20 44 72
7084 69 76 65
7087 20 4E 61
708A 6D 65 20
708D 3D 20
708F 00          62          DEFB    0
7090 CD C6 72    63          CALL    DSKGET
7093 D8          64          RET     C
7094 CD E9 72    65          CALL    SURE
7097 D8          66          RET     C
7098 20 DC       67          JR      NZ,SYSRD1
709A CD E2 1F    68 SYSRD2   CALL    #MPRNT
709D 0D 0D       69          DEFB    0DH:0DH
709F 4E 6F 77    70          DEFM    "Now System Loading...."
70A2 20 53 79
70A5 73 74 65
70A8 6D 20 4C
70AB 6F 61 64
70AE 69 6E 67
70B1 2E 2E 2E
70B4 2E
70B5 00          71          DEFB    0
70B6 21 00 80    72          LD      HL,8000H
70B9 11 00 00    73          LD      DE,0
70BC 3E 01       74          LD      A,1
70BE CD 00 20    75          CALL    #DRDSB
70C1 DA 18 73    76          JP      C,ERROR
70C4 21 00 81    77          LD      HL,8100H
70C7 11 20 00    78          LD      DE,20H
70CA 3E 28       79          LD      A,40
70CC CD 00 20    80          CALL    #DRDSB
70CF DA 18 73    81          JP      C,ERROR
70D2             82
70D2 CD 55 73    83          CALL    CHKSYS           ; SYSTI
70D5 28 26       84          JR      Z,RDEND
70D7 CD C4 1F    85          CALL    #BELL
70DA CD E2 1F    86          CALL    #MPRNT
70DD 0D 0D       87          DEFB    0DH:0DH
70DF 54 68 69    88          DEFM    "This is not System Disk
70E2 73 20 69
70E5 73 20 6E
70E8 6F 74 20
70EB 53 79 73
70EE 74 65 6D
70F1 20 44 69
70F4 73 6B 20
70F7 21 21
70F9 00          89          DEFB    0
70FA C3 76 70    90          JP      SYSRD1
70FD             91
70FD             92 RDEND:
70FD 3E FF       93          LD      A,0FFH
70FF 32 6B 73    94          LD      (LOAD),A
7102             95          ;
7102 CD C4 1F    96 SYSWT1   CALL    #BELL
7105 CD E2 1F    97          CALL    #MPRNT
7108 0D 0D       98          DEFB    0DH:0DH
710A 53 79 73    99          DEFM    "System Load Complete !!"
710D 74 65 6D
7110 20 4C 6F
7113 61 64 20
7116 43 6F 6D
7119 70 6C 65
711C 74 65 20
711F 21 21
7121 0D 0D 00   100          DEFB    0DH:0DH:0
7124            101
7124            102 ;        SELECT DISTINATION DRIVE
7124            103
7124 CD E2 1F   104 SYSWT2   CALL    #MPRNT
7127 0D 0D      105          DEFB    0DH:0DH
7129 44 65 73   106          DEFM    "Destination Drive Name
712C 74 69 6E
712F 61 74 69
7132 6F 6E 20
7135 44 72 69
7138 76 65 20
713B 4E 61 6D
713E 65 20 3D
7141 20
7142 00         107          DEFB    0
7143 CD C6 72   108          CALL    DSKGET
7146 D8         109          RET     C
7147 CD E9 72   110          CALL    SURE
714A D8         111          RET     C
714B 20 B5      112          JR      NZ,SYSWT1
714D            113
714D            114 ;        CHECK SYSTEM ?
714D            115
714D ED 5B 5E   116          LD      DE,(#FATPOS)
7150 1F
7151 2A 62 1F   117          LD      HL,(#FATBF)
7154 3E 01      118          LD      A,1
7156 CD 00 20   119          CALL    #DRDSB
7159 DA 18 73   120          JP      C,ERROR
715C 11 F5 73   121          LD      DE,FATDTA
715F 1A         122 LPFT     LD      A,(DE)
7160 B7         123          OR      A
7161 28 61      124          JR      Z,ITSSYS
7163 13         125          INC     DE
7164 BE         126          CP      (HL)
7165 23         127          INC     HL
7166 20 02      128          JR      NZ,CANWT
7168 18 F5      129          JR      LPFT
716A            130
716A            131 ;        WRITE SYSTEM
716A            132
716A CD E2 1F   133 CANWT    CALL    #MPRNT
716D 0D 0D      134          DEFB    0DH:0DH
716F 4E 6F 77   135          DEFM    "Now System Writing....'
7172 20 53 79
7175 73 74 65
7178 6D 20 57
717B 72 69 74
717E 69 6E 67
7181 2E 2E 2E
7184 2E
7185 0D 00      136          DEFB    0DH:0
7187 21 00 80   137          LD      HL,8000H
718A 11 00 00   138          LD      DE,0
718D 3E 01      139          LD      A,1
718F CD 03 20   140          CALL    #DWTSB
7192 DA 18 73   141          JP      C,ERROR
7195 21 00 81   142          LD      HL,8100H
7198 11 20 00   143          LD      DE,20H
719B 3E 28      144          LD      A,40
719D CD 03 20   145          CALL    #DWTSB
71A0 DA 18 73   146          JP      C,ERROR
71A3 CD E2 1F   147          CALL    #MPRNT
71A6 0D         148          DEFB    0DH
71A7 53 79 73   149          DEFM    "System Generate Complet
71AA 74 65 6D
71AD 20 47 65
71B0 6E 65 72
71B3 61 74 65
71B6 20 43 6F
71B9 6D 70 6C
71BC 65 74 65
71BF 20 21
71C1 00         150          DEFB    0
71C2 18 6F      151          JR      FORMAT2
71C4            152 ;
71C4            153 ITSSYS:
71C4 CD C4 1F   154          CALL    #BELL
71C7 CD E2 1F   155          CALL    #MPRNT
71CA 0D 0D      156          DEFB    0DH:0DH
71CC 44 65 73   157          DEFM    "Destination is System Di
71CF 74 69 6E
71D2 61 74 69
71D5 6F 6E 20
71D8 69 73 20
71DB 53 79 73
71DE 74 65 6D
71E1 20 44 69
71E4 73 6B 2E
71E7 0D         158          DEFB    0DH
71E8 44 6F 20   159          DEFM    "Do you want to write on
71EB 79 6F 75
71EE 20 77 61
71F1 6E 74 20
71F4 74 6F 20
71F7 77 72 69
71FA 74 65 20
71FD 6F 6E 20
7200 69 74 20
7203 3F
7204 00         160          DEFB    0
7205 CD E9 72   161          CALL    SURE
7208 D8         162          RET     C
7209 C2 24 71   163          JP      NZ,SYSWT2
720C C3 6A 71   164          JP      CANWT
720F            165          ;
720F            166
720F            167 ;        LOGICAL FORMAT
720F            168
720F CD E2 1F   169 FORMAT1  CALL    #MPRNT
7212 0D 0D      170          DEFB    0DH:0DH
7214 46 6F 72   171          DEFM    "Format Drive Name = "
7217 6D 61 74
721A 20 44 72
721D 69 76 65
7220 20 4E 61
7223 6D 65 20
7226 3D 20
7228 00         172          DEFB    0
7229 CD C6 72   173          CALL    DSKGET
722C D8         174          RET     C
722D CD E9 72   175          CALL    SURE
7230 D8         176          RET     C
7231 20 DC      177          JR      NZ,FORMAT1
7233 CD E2 1F   178 FORMAT2  CALL    #MPRNT
7236 0D 0D      179          DEFB    0DH:0DH
7238 49 6E 69   180          DEFM    "Initializing..."
723B 74 69 61
723E 6C 69 7A
7241 69 6E 67
7244 2E 2E 2E
7247 0D 00      181          DEFB    0DH:0
7249 3E 01      182          LD      A,1
724B ED 5B 62   183          LD      DE,(#FATBF)
724E 1F
724F 12         184          LD      (DE),A
7250 13         185          INC     DE
7251 3E 8F      186          LD      A,8FH
7253 12         187          LD      (DE),A
7254 13         188          INC     DE
7255 0E 03      189          LD      C,3
7257 3A 6C 73   190          LD      A,(WORKNO)
725A FE 31      191          CP      '1'
725C 28 0D      192          JR      Z,FORMAT3
725E            193          ;
725E            194
725E            195 ;        MAKE FAT
725E            196 ;
725E 3E 03      197          LD      A,3              ;If System Generated
```


▶学校の実験で，8080を載せたワンボードマイコンを使ったアセンブリ言語の実習があった。なんて古くさい石を使っているのだと思ったが，8080のアセンブリ言語の記述の素晴しさを改めて認識しました。わかりますか？　福島　淑生（22）鹿児島県

```
7260 12          198          LD      (DE),A
7261 13          199          INC     DE
7262 3C          200          INC     A
7263 12          201          LD      (DE),A
7264 13          202          INC     DE
7265 3E 8F       203          LD      A,8FH
7267 12          204          LD      (DE),A
7268 13          205          INC     DE
7269 0E 06       206          LD      C,6
726B             207          ;
726B AF          208 FORMAT3  XOR     A
726C 12          209          LD      (DE),A
726D 62 6B       210          LD      HL,DE
726F 13          211          INC     DE
7270 3A 66 1F    212          LD      A,(#MXTRK)
7273 91          213          SUB     C
7274 4F          214          LD      C,A
7275 06 00       215          LD      B,0
7277 ED B0       216          LDIR
7279 3E 8F       217          LD      A,8FH
727B 23          218          INC     HL
727C 13          219          INC     DE
727D 77          220          LD      (HL),A
727E 3A 66 1F    221          LD      A,(#MXTRK)
7281 4F          222          LD      C,A
7282 3E FF       223          LD      A,0FFH
7284 91          224          SUB     C
7285 4F          225          LD      C,A
7286 06 00       226          LD      B,0
7288 ED B0       227          LDIR
728A 3E 01       228          LD      A,1
728C ED 5B 5E    229          LD      DE,(#FATPOS)
728F 1F
7290 2A 62 1F    230          LD      HL,(#FATBF)
7293 CD 03 20    231          CALL    #DWTSB
7296 DA 18 73    232          JP      C,ERROR
7299             233
7299             234 ;        MAKE DIRECTORY
7299             235
7299 21 00 B0    236          LD      HL,0B000H
729C 36 FF       237          LD      (HL),0FFH
729E 11 01 B0    238          LD      DE,0B001H
72A1 01 FF 0F    239          LD      BC,0FFFH
72A4 ED B0       240          LDIR
72A6 3E 10       241          LD      A,10H
72A8 ED 5B 60    242          LD      DE,(#DIRPS)
72AB 1F
72AC 21 00 B0    243          LD      HL,0B000H
72AF CD 03 20    244          CALL    #DWTSB
72B2 38 64       245          JR      C,ERROR
72B4             246          ;
72B4 CD E2 1F    247          CALL    #MPRNT
72B7 0D          248          DEFB    0DH
72B8 43 6F 6D    249          DEFM    "Complete !"
72BB 70 6C 65
72BE 74 65 20
72C1 21
72C2 0D 00       250          DEFB    0DH:0
72C4 18 72       251          JR      AGAIN
72C6             252
72C6             253
72C6             254 ;        GET DRIVE NAME
72C6             255 ;
72C6             256 ;        Cy=1 : BREAK KEY PRESSED
72C6             257
72C6 CD 21 20    258 DSKGET   CALL    #FLGET
72C9 FE 1B       259          CP      1BH
72CB 20 02       260          JR      NZ,DSKGT1
72CD 37          261          SCF                     ;Break Key Pressed
72CE C9          262          RET
72CF FE 61       263 DSKGT1   CP      'a'
72D1 38 06       264          JR      C,DSKGT2
72D3 FE 65       265          CP      'd'+1
72D5 30 EF       266          JR      NC,DSKGET
72D7 D6 20       267          SUB     20H
72D9 FE 41       268 DSKGT2   CP      'A'
72DB 38 E9       269          JR      C,DSKGET
72DD FE 45       270          CP      'D'+1
72DF 30 E5       271          JR      NC,DSKGET
72E1 32 5D 1F    272          LD      (#DSK),A
72E4 CD F4 1F    273          CALL    #PRINT
72E7 B7          274          OR      A
72E8 C9          275          RET
72E9             276
72E9             277 ;
72E9             278 ;        ASK SURE ?
72E9             279 ;
72E9             280 ;        CY=1 : BREAK KEY PRESSED
72E9             281 ;        Z=1  : OK.
72E9             282 ;
72E9             283
72E9 CD E2 1F    284 SURE     CALL    #MPRNT
72EC 0D 0D       285          DEFB    0DH,0DH
72EE 41 72 65    286          DEFM    "Are you sure (Y/N) ? "
72F1 20 79 6F
72F4 75 20 73
72F7 75 72 65
72FA 20 28 59
72FD 2F 4E 29
7300 20 3F 20
7303 00          287          DEFB    0
7304 CD 21 20    288 SURE0    CALL    #FLGET
7307 FE 1B       289          CP      1BH
7309 20 02       290          JR      NZ,SURE1
730B 37          291          SCF                     ;Break Key Pressed
730C C9          292          RET
730D CD F4 1F    293 SURE1    CALL    #PRINT
7310 FE 59       294          CP      'Y'
7312 C8          295          RET     Z
7313 FE 79       296          CP      'y'
7315 37          297          SCF
7316 3F          298          CCF
7317 C9          299          RET
7318             300
7318             301 ;
7318             302 ;        ERROR
7318             303 ;        PRINT ERROR MESSAGE
7318             304
7318 CD EE 1F    305 ERROR    CALL    #LTNL
731B CD 33 20    306          CALL    #ERROR
731E CD E2 1F    307          CALL    #MPRNT
7321 52 65 74    308          DEFM    "Retry (Y/N) ? "
7324 72 79 20
7327 28 59 2F
732A 4E 29 20
732D 3F 20
732F 00          309          DEFB    0
7330 CD 04 73    310          CALL    SURE0
7333 D8          311          RET     C
7334 CA 00 70    312          JP      Z,RETRY
7337 C9          313          RET
7338             314
7338             315 ;        ASK AGEIN
7338             316 ;        Y : GOTO MENU
7338             317 ;        N : RETURN TO SYSTEM
7338             318 ;
7338             319
7338 CD E2 1F    320 AGAIN    CALL    #MPRNT
733B 0D          321          DEFB    0DH
733C 41 67 61    322          DEFM    "Again (Y/N) ? "
733F 69 6E 20
7342 28 59 2F
7345 4E 29 20
7348 3F 20
734A 00          323          DEFB    0
734B CD 04 73    324          CALL    SURE0
734E D8          325          RET     C
734F CA 04 70    326          JP      Z,AGAIN0
7352 C3 56 70    327          JP      ENDFMT
7355             328
7355             329 ;        CHECK SYSTEM DISK
7355             330 ;
7355             331 ;        Z=1 : SYSTEM DISK
7355             332 ;        Z=0 : NOT SYSTEM DISK
7355             333
7355             334 CHKSYS:
7355 21 6D 73    335          LD      HL,CHKDTA
7358 0E 04       336          LD      C,4
735A             337 CHKNXT:
735A 5E          338          LD      E,(HL)
735B 23          339          INC     HL
735C 56          340          LD      D,(HL)
735D 23          341          INC     HL
735E 06 20       342          LD      B,32
7360 1A          343 CHKLOOP  LD      A,(DE)
7361 13          344          INC     DE
7362 BE          345          CP      (HL)
7363 23          346          INC     HL
7364 C0          347          RET     NZ
7365 10 F9       348          DJNZ    CHKLOOP
7367 0D          349          DEC     C
7368 20 F0       350          JR      NZ,CHKNXT
736A C9          351          RET
736B             352          ;
736B 00          353 LOAD     DEFS    1
736C 00          354 WORKNO   DEFS    1
736D             355
736D             356 CHKDTA:
736D 10 80       357          DEFW    8010H     ;0800H (IPL)
736F 20 53 2D    358          DEFM    " S-OS SWORD IPL "
7372 4F 53 20
7375 53 57 4F
7378 52 44 20
737B 49 50 4C
737E 20
737F 73 74 61    359          DEFM    "started !"
7382 72 74 65
7385 64 20 21
7388 0D 00 07    360          DB      0DH:00H:07H
738B 2A 2A 2A    361          DEFM    "****"
738E 2A
738F             362
738F 00 9A       363          DEFW    9A00H     ;2100H (DOS MODULE)
7391 ED 7B 6C    364          DEFB    0EDH:07BH:06CH:01FH:0CDH:0D6H:01FH:03EH
7394 1F CD D6
7397 1F 3E
7399 23 CD F4    365          DEFB    023H:0CDH:0F4H:01FH:0EDH:05BH:076H:01FH
739C 1F ED 5B
739F 76 1F
73A1 CD D3 1F    366          DEFB    0CDH:0D3H:01FH:0CDH:01BH:021H:0DCH:033H
73A4 CD 1B 21
73A7 DC 33
73A9 20 18 E5    367          DEFB    020H:018H:0E5H:01AH:0FEH:023H:028H:002H
73AC 1A FE 23
73AF 28 02
73B1             368
73B1 00 9F       369          DEFW    9F00H     ;2600 (DOS MODULE)
73B3 7E 32 DE    370          DEFB    07EH:032H:0DEH:027H:0EBH:029H:029H:029H
73B6 27 EB 29
73B9 29 29
73BB 29 EB E1    371          DEFB    029H:0EBH:0E1H:0B7H:028H:019H:0FEH:080H
73BE B7 28 19
73C1 FE 80
73C3 30 19 3E    372          DEFB    030H:019H:03EH:010H:0CDH:044H:025H:0D8H
73C6 10 CD 44
73C9 25 D8
73CB 11 00 10    373          DEFB    011H:000H:010H:019H:0E5H:069H:060H:0B7H
73CE 19 E5 69
73D1 60 B7
73D3             374
73D3 80 A3       375          DEFW    0A380H    ;2A80 (DOS MODULE)
73D5 6F 63 61    376          DEFB    06FH:063H:061H:074H:069H:06FH:06EH:020H
73D8 74 69 6F
73DB 6E 20
73DD 54 61 62    377          DEFB    054H:061H;062H:06CH:065H:00DH:046H:069H
73E0 6C 65 0D
73E3 46 69
73E5 6C 65 20    378          DEFB    06CH:065H:020H:06EH:06FH:074H:020H:046H
73E8 6E 6F 74
73EB 20 46
73ED 6F 75 6E    379          DEFB    06FH:075H:06EH:064H:00DH:044H:065H:076H
73F0 64 0D 44
73F3 65 76
73F5             380
73F5             381 FATDTA:
73F5 01 8F 03    382          DEFB    01H:8FH:03H:04H:8FH:0
73F8 04 8F 00
```

リスト18　カード版デバッガヘルプメニュー

```
1435 1143   23   DHLPM  FCC  '################################################################'
1436 117B   0D          FCB  CR
1437 117C   23          FCC  '#      S-OS SWORD/FM-7 Z-80 Debugger command menu      #'
1438 11B4   0D          FCB  CR
1439 11B5   23          FCC  '################################################################'
1440 11ED   0D          FCB  CR
1441 11EE   23          FCC  '# B                  # Display current break point.      #'
1442 1226   0D          FCB  CR
1443 1227   23          FCC  '# B *                # Release current break point.      #'
1444 125F   0D          FCB  CR
1445 1260   23          FCC  '# B adr1             # Set break point on (adr1).        #'
1446 1298   0D          FCB  CR
1447 1299   23          FCC  '# C adr1,adr2        # Caliculate adr1+adr2,adr1-adr2.   #'
1448 12D1   0D          FCB  CR
1449 12D2   23          FCC  '# D[adr1[,adr2]]     # Dump from adr1 to adr2.           #'
1450 130A   0D          FCB  CR
1451 130B   23          FCC  '# F adr1,adr2,dt     # Fill from adr1 to adr2 with dt.   #'
1452 1343   0D          FCB  CR
1453 1344   23          FCC  '# G[adr1]            # Go object program.                #'
1454 137C   0D          FCB  CR
1455 137D   23          FCC  '# H                  # Display help message.             #'
1456 13B5   0D          FCB  CR
1457 13B6   23          FCC  '# I adr1,letter      # Insert memory stamp on (adr1)-.   #'
1458 13EE   0D          FCB  CR
1459 13EF   23          FCC  '# J adr1             # Jump to adr1.                     #'
1460 1427   0D          FCB  CR
1461 1428   23          FCC  '# M[adr1]            # Exchange contents of memory.      #'
1462 1460   0D          FCB  CR
1463 1461   23          FCC  '# P                  # Complement printer switch.        #'
1464 1499   0D          FCB  CR
1465 149A   23          FCC  '# Q                  # Quit to SWORD.                    #'
1466 14D2   0D          FCB  CR
1467 14D3   23          FCC  '# R                  # Display/exchange registers.       #'
1468 150B   0D          FCB  CR
1469 150C   23          FCC  '# S adr1,adr2,dt     # Serch "dt" from adr1 to adr2.     #'
1470 1544   0D          FCB  CR
1471 1545   23          FCC  '# T adr1,adr2,adr3 # Transfer from adr1-adr2 to adr3.   #'
1472 157D   0D          FCB  CR
1473 157E   23          FCC  '################################################################'
1474 15B6   0D          FCB  CR,EOS
```

▶「BASICで数学と遊ぶ」では，数学と離れて久しい私にとって，思い出すのに１日タバコ50本を必要としますが，懐かしくてついつい読んでしまいます。

批杷　利政（27）青森県

X1/X1turbo用

パズルゲーム

STAR PANIC

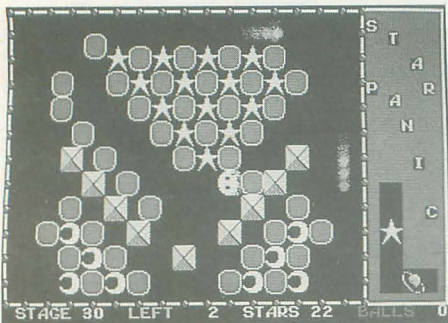
Nakaoka Toshihiro
中岡　敏博

BGMも軽快なX1用パズルゲームの登場だ。モンスターに盗まれた星を求めて不思議な世界を探険しよう。だけど無重力空間では動きもままならない。うまくコツをつかんで30面クリアをめざすのだ。

ストーリー

ある日突然，夜空から月やお星さまが消えてしまいました。別の世界からやってきたコメットや変てこりんなモンスターたちにすべて持ち去られてしまったのです。奪われた星を取り戻すため主人公は不思議な世界へと向かったのですが……。ということで，このプログラムは画面上の星を集めてまわるというアクション風のパズルゲームです。面は全部で30面，星を守って動きまわるモンスターたちに気をつけて夜空に星を取り戻してください。

入力方法

X1用のBASIC（turbo BASIC不可）であればどのバージョンでも動作します。まず，BASICを起動してリスト1，2を入力し，続いてMACINTO-Cなどのマシン語入力ツールからリスト3を打ち込みます。なお，リスト2，610行以下のデータ部分で，最初から2文字目の空白はスペースではありません。カナ＋シフト＋ロ（キーボード右下隅）のキーを押してください。リスト2，3はそれぞれ，

リスト2　“STAR MAIN”
リスト3　“STAR MON”

のファイルネームでプログラム中から呼び出されていますので，セーブする際はファイルネームを間違えないように注意してください。

操作方法と基本ルール

テンキーの2,8,4,6で上下左右にキャラクターを動かしてください。ゲームの目的は画面上に散らばった星や月や太陽を拾い集めることです。しかし，これらは横方向からしか取ることはできません。また，面によってはコメットなどのモンスターが星を守っており，ガラスの刃のようなブロックに触れると体がはじけて死んでしまいます。当然，モンスターに捕まっても主人公は死んでしまいますが，こちらからモンスターたちを殺すことはできません。

主人公の足には特殊な吸盤が付いており，敵キャラクター，青いガラスの刃のようなブロック以外の物の上に張りついて移動することができます。この世界には重力はありませんが，星，赤いブロックなどのキャラクターを両足で踏むと，そのときの主人公に対して相対的な下方向に落下を始めます。なお，落下中でも移動したり，ジャンプしたりすることは可能です。

曲がり角などで体が半分はみだした状態では，それまで進んでいた方向に進むキーを1回余分に押すと落っこちずに角を曲がることができます。また，スペースキーを押すとジャンプでき，障害物の上に体の向きが上下逆になった状態で着地します。ゲーム中，どうしてもクリアできない状況に陥ったときはブレイクキーを押してください（シフト＋ブレイクではない）。

プログラムについて

主人公の人数を増やしたいときはリスト2の150行のLTの値を変えてください。同じく150行のMには面数－1の値が格納されていますから，1面目以外の面からスタートしたいときにはここを変更すればよいでしょう。このあたりは各自のお好きなようにしてください。

マシン語部分のうちBA00HからCD5FHまでの部分は各面のデータになっています。データは1バイト2キャラクタの構成で，1面は36×18（324バイト）が30面分用意してあります。画面と1対1に対応しているので解析や変更もそう困難ではないでしょう。

そのほか，CD60HからED5FHまでがキャラクターデータ，メインプログラムはED60HからFDFFHまでとなっていますので，入力の際はとくに後半に注意して打ち込んでください。それでは夜空の旅をお楽しみください。

Profile
◇中岡君は広島県にお住まいの17歳，現在高校3年生です。マイコン歴は約3年，使用機種はX1Ckで今度はグラフィック関係のルーチンを自作したいそうです。

リスト1　STAR PANIC

```
10 '////////////////////
20 '/                  /
30 '/   PUZZLE GAME    /
40 '/                  /
50 '/  "STAR   PANIC"  /
60 '/                  /
70 '/                  /
80 '////////////////////
90 '
100 WIDTH 40:INIT:CLS4
110 '
120 LINE(32,88)-(281,112),PSET,1,BF
130 LINE(32,88)-(281,112),PSET,7,B
140 LOCATE 5,12:PRINT"Now PCG Setting. Please Wait."
150 GOSUB "PCGSET"
160 CLS4
170 '
180 SCREEN0,1,0:CGEN1:LOCATE 0,0
190 PRINT"
200 PRINT"
210 PRINT"
220 PRINT"
230 PRINT"
240 PRINT"
250 PRINT"
260 PRINT"
270 PRINT"
280 PRINT"
290 PRINT
300 PRINT"
310 PRINT"
320 PRINT"
330 PRINT"
340 PRINT"
350 PRINT"
360 PRINT"
370 PRINT"
380 PRINT"
390 PRINT"
400 PRINT
410 PRINT"      PROGRAM BY TOSHIHIRO NAKAOKA"
420 '
430 SCREEN0,0:COLOR7
440 LOCATE 1,4:PRINT"MOVE .... 4826 KEY"
450 LOCATE 20,4:PRINT"JUMP .... SPACE KEY"
460 LOCATE 6,8 :CSIZE2:PRINT#0,"STARS"
470 LOCATE 5,10:PRINT" 。 「」 、・ ヲァ"
480 LOCATE 5,11:PRINT"-ア イウ エオ カキ"
490 LOCATE 26,8 :PRINT#0,"BALLS"
500 LOCATE 25,10:PRINT"ィウ エオ ャユ ョッ"
510 LOCATE 25,11:PRINT"クケ コサ シス セソ"
520 LOCATE 18,8:PRINT"STONE"
530 LOCATE 20,10:PRINT"__ー|"
540 PALET 0,2:PALET 1,0
550 LINE(0,16)-(319,104),PSET,1,BF
560 LOCATE 8,18:PRINT#0,"NOW LOADING"
570 LOCATE 8,20:PRINT#0,"PLEASE  WAIT"
580 KEY5,"RUN 20"+CHR$(13):RUN"STAR MAIN"
590 LABEL "PCGSET"
600 DEFCHR$(0)=HEXCHR$("000000000000000000000000000000000000000000000000")
610 DEFCHR$(1)=HEXCHR$("000000000000000000000000000000000000000000000000")
620 DEFCHR$(2)=HEXCHR$("000000000000000000000000000000000000000000000000")
630 DEFCHR$(3)=HEXCHR$("000000000000000000000000000000000000000000000000")
640 DEFCHR$(4)=HEXCHR$("000000000000000000000000000000000000000000000000")
650 DEFCHR$(5)=HEXCHR$("000000000000000000000000000000000000000000000000")
660 DEFCHR$(6)=HEXCHR$("000000000000000000000000000000000000000000000000")
670 DEFCHR$(7)=HEXCHR$("000000000000000000000000000000000000000000000000")
680 DEFCHR$(8)=HEXCHR$("10387CFEFFFFFFFF100804020101010110387CFE7F7F7F7F")
690 DEFCHR$(9)=HEXCHR$("10387CFEFFFFFFFF100804020101010110387CFE7F7F7F7F")
700 DEFCHR$(10)=HEXCHR$("0F1F3F7FFF7F3F1F0F102040800000000F1F3F7FFF7F3F10")
710 DEFCHR$(11)=HEXCHR$("FFFFFFFFFFFFFFFF0000000000000000FFFFFFFFFFFFFF00")
720 DEFCHR$(12)=HEXCHR$("FFFFFFFFFFFFFFFF0000000000000000FFFFFFFFFFFFFF00")
730 DEFCHR$(13)=HEXCHR$("F0F8FCFEFFFEFCF8F008040201000000F0F8FCFEFFFEFC08")
740 DEFCHR$(14)=HEXCHR$("FFFFFFFFFFFFFFFF01010101010101017F7F7F7F7F7F7F7F")
750 DEFCHR$(15)=HEXCHR$("FFFFFFFFFFFFFFFF01010101010101017F7F7F7F7F7F7F7F")
760 DEFCHR$(16)=HEXCHR$("000000000000000000000000000000000000000000000000")
770 DEFCHR$(17)=HEXCHR$("000000000000000000000000000000000000000000000000")
780 DEFCHR$(18)=HEXCHR$("000000000000000000000000000000000000000000000000")
790 DEFCHR$(19)=HEXCHR$("000000000000000000000000000000000000000000000000")
800 DEFCHR$(20)=HEXCHR$("000000000000000000000000000000000000000000000000")
810 DEFCHR$(21)=HEXCHR$("000000000000000000000000000000000000000000000000")
820 DEFCHR$(22)=HEXCHR$("000000000000000000000000000000000000000000000000")
830 DEFCHR$(23)=HEXCHR$("000000000000000000000000000000000000000000000000")
840 DEFCHR$(24)=HEXCHR$("FFFFFFFFFFFFFFFF01010101010101017F7F7F7F7F7F7F7F")
850 DEFCHR$(25)=HEXCHR$("FFFFFFFFFFFFFFFF01010101010101017F7F7F7F7F7F7F7F")
860 DEFCHR$(26)=HEXCHR$("0F1F3F7FFF7F3F1F0F102040800000000F1F3F7FFF7F3F10")
870 DEFCHR$(27)=HEXCHR$("FFFFFFFFFFFFFFFF0000000000000000FFFFFFFFFFFFFF00")
880 DEFCHR$(28)=HEXCHR$("FFFFFFFFFFFFFFFF0000000000000000FFFFFFFFFFFFFF00")
890 DEFCHR$(29)=HEXCHR$("F0F8FCFEFFFEFCF8F008040201000000F0F8FCFEFFFEFC08")
900 DEFCHR$(30)=HEXCHR$("FFFFFFFFFE7C381001010101020408107F7F7F7FFE7C3810")
910 DEFCHR$(31)=HEXCHR$("FFFFFFFFFE7C381001010101020408107F7F7F7FFE7C3810")
920 DEFCHR$(32)=HEXCHR$("000000000000000000000000000000000000000000000000")
930 DEFCHR$(33)=HEXCHR$("181818181800180018181818180018001818181818001800")
940 DEFCHR$(34)=HEXCHR$("6C6C3600000000006C6C3600000000006C6C360000000000")
950 DEFCHR$(35)=HEXCHR$("36366C000000000036366C000000000036366C0000000000")
960 DEFCHR$(36)=HEXCHR$("183E583C1A7C1800183E583C1A7C1800183E583C1A7C1800")
970 DEFCHR$(37)=HEXCHR$("0063660C183666000063660C183666000063660C18366600")
980 DEFCHR$(38)=HEXCHR$("386C6C396A663B00386C6C396A663B00386C6C396A663B00")
990 DEFCHR$(39)=HEXCHR$("181808100000000018180810000000001818081000000000")
```

```
1000 DEFCHR$(40)=HEXCHR$("081030303010080008103030301008000810303030100800")
1010 DEFCHR$(41)=HEXCHR$("201018181810200020101818181020002010181818102000")
1020 DEFCHR$(42)=HEXCHR$("1054387C385410001054387C385410001054387C38541000")
1030 DEFCHR$(43)=HEXCHR$("0018187E181800000018187E181800000018187E18180000")
1040 DEFCHR$(44)=HEXCHR$("000000003818300000000000381830000000000038183000")
1050 DEFCHR$(45)=HEXCHR$("0000007E000000000000007E000000000000007E00000000")
1060 DEFCHR$(46)=HEXCHR$("000000003838000000000000383800000000000038380000")
1070 DEFCHR$(47)=HEXCHR$("0003060C183060000003060C183060000003060C18306000")
1080 DEFCHR$(48)=HEXCHR$("3C66666666663C003C66666666663C003C66666666663C00")
1090 DEFCHR$(49)=HEXCHR$("181818181818180018181818181818001818181818181800")
1100 DEFCHR$(50)=HEXCHR$("3C66061C30607E003C66061C30607E003C66061C30607E00")
1110 DEFCHR$(51)=HEXCHR$("3C66063C06663C003C66063C06663C003C66063C06663C00")
1120 DEFCHR$(52)=HEXCHR$("060E1E36667F0600060E1E36667F0600060E1E36667F0600")
1130 DEFCHR$(53)=HEXCHR$("7E607C0602663C007E607C0602663C007E607C0602663C00")
1140 DEFCHR$(54)=HEXCHR$("1C30607C66663C001C30607C66663C001C30607C66663C00")
1150 DEFCHR$(55)=HEXCHR$("7E660C18181818007E660C18181818007E660C1818181800")
1160 DEFCHR$(56)=HEXCHR$("3C66663C66663C003C66663C66663C003C66663C66663C00")
1170 DEFCHR$(57)=HEXCHR$("3C66663E060C38003C66663E060C38003C66663E060C3800")
1180 DEFCHR$(58)=HEXCHR$("003030003030000000303000303000000030300030300000")
1190 DEFCHR$(59)=HEXCHR$("003030003010200000303000301020000030300030102000")
1200 DEFCHR$(60)=HEXCHR$("0C18306030180C000C18306030180C000C18306030180C00")
1210 DEFCHR$(61)=HEXCHR$("00007E007E00000000007E007E00000000007E007E000000")
1220 DEFCHR$(62)=HEXCHR$("30180C060C18300030180C060C18300030180C060C183000")
1230 DEFCHR$(63)=HEXCHR$("3C66060C180018003C66060C180018003C66060C18001800")
1240 DEFCHR$(64)=HEXCHR$("1E214D554E211E001E214D554E211E001E214D554E211E00")
1250 DEFCHR$(65)=HEXCHR$("070B13237F636300070B13237F636300070B13237F636300")
1260 DEFCHR$(66)=HEXCHR$("7C66667C63637E007C66667C63637E007C66667C63637E00")
1270 DEFCHR$(67)=HEXCHR$("1E23606060231E001E23606060231E001E23606060231E00")
1280 DEFCHR$(68)=HEXCHR$("7C26232323267C007C26232323267C007C26232323267C00")
1290 DEFCHR$(69)=HEXCHR$("7E60607C60607F007E60607C60607F007E60607C60607F00")
1300 DEFCHR$(70)=HEXCHR$("7F60607C606060007F60607C606060007F60607C60606000")
1310 DEFCHR$(71)=HEXCHR$("1E23606067211E001E23606067211E001E23606067211E00")
1320 DEFCHR$(72)=HEXCHR$("6363637F636363006363637F636363006363637F63636300")
1330 DEFCHR$(73)=HEXCHR$("3C18181818183C003C18181818183C003C18181818183C00")
1340 DEFCHR$(74)=HEXCHR$("0F06060666663C000F06060666663C000F06060666663C00")
1350 DEFCHR$(75)=HEXCHR$("63666C786666630063666C786666630063666C7866666300")
1360 DEFCHR$(76)=HEXCHR$("6060606060607F006060606060607F006060606060607F00")
1370 DEFCHR$(77)=HEXCHR$("63776B6B6363630063776B6B6363630063776B6B63636300")
1380 DEFCHR$(78)=HEXCHR$("4363736B676363004363736B676363004363736B67636300")
1390 DEFCHR$(79)=HEXCHR$("1C36636363361C001C36636363361C001C36636363361C00")
1400 DEFCHR$(80)=HEXCHR$("7E6363637E6060007E6363637E6060007E6363637E606000")
1410 DEFCHR$(81)=HEXCHR$("1C2663636B3E1B001C2663636B3E1B001C2663636B3E1B00")
1420 DEFCHR$(82)=HEXCHR$("7E6363637E6C63007E6363637E6C63007E6363637E6C6300")
1430 DEFCHR$(83)=HEXCHR$("3E63603E03633E003E63603E03633E003E63603E03633E00")
1440 DEFCHR$(84)=HEXCHR$("7E181818181818007E181818181818007E18181818181800")
1450 DEFCHR$(85)=HEXCHR$("6363636363633E006363636363633E006363636363633E00")
1460 DEFCHR$(86)=HEXCHR$("63636336361C080063636336361C080063636336361C0800")
1470 DEFCHR$(87)=HEXCHR$("6363636B6B7763006363636B6B7763006363636B6B776300")
1480 DEFCHR$(88)=HEXCHR$("6363361C366363006363361C366363006363361C36636300")
1490 DEFCHR$(89)=HEXCHR$("666666663C181800666666663C181800666666663C181800")
1500 DEFCHR$(90)=HEXCHR$("3F460C1830617E003F460C1830617E003F460C1830617E00")
1510 DEFCHR$(91)=HEXCHR$("3C20202020203C003C20202020203C003C20202020203C00")
1520 DEFCHR$(92)=HEXCHR$("66247E187E18180066247E187E18180066247E187E181800")
1530 DEFCHR$(93)=HEXCHR$("780808080808780078080808080878007808080808087800")
1540 DEFCHR$(94)=HEXCHR$("183C660000000000183C660000000000183C660000000000")
1550 DEFCHR$(95)=HEXCHR$("000000000000FF00000000000000FF00000000000000FF00")
1560 DEFCHR$(96)=HEXCHR$("7F3F1F0F070301007F3F1F0F070301007FBFDFEFF7FBFDFE")
1570 DEFCHR$(97)=HEXCHR$("FEFCF9F2E5CA952AFEFCF9F2E5CA952AFEFDFBF7EFDFBF7F")
1580 DEFCHR$(98)=HEXCHR$("000000000000000000000000000000007F7F7F7F7F7F7F7F")
1590 DEFCHR$(99)=HEXCHR$("54AA54AA54AA54AA54AA54AA54AA54AAFEFEFEFEFEFEFEFE")
1600 DEFCHR$(100)=HEXCHR$("00FFFFFFFFFFFFFF00FFFFFFFFFFFFFF00FFFFFFFFFFFFFF")
1610 DEFCHR$(101)=HEXCHR$("00FFFFFFFFFFFFFF00FFFFFFFFFFFFFF00FFFFFFFFFFFFFF")
1620 DEFCHR$(102)=HEXCHR$("00000000000000000000000000000000FFFFFFFFFFFFFFFF")
1630 DEFCHR$(103)=HEXCHR$("00000000000000000000000000000000FFFFFFFFFFFFFFFF")
1640 DEFCHR$(104)=HEXCHR$("201700000020202060F7F760006060602017070000606060")
1650 DEFCHR$(105)=HEXCHR$("02E100000002020206EFEF060006060602E1E00000060606")
1660 DEFCHR$(106)=HEXCHR$("20202020200020106060606060600060F060606060002010")
1670 DEFCHR$(107)=HEXCHR$("020202020200020106060606060600060F06060606000201")
1680 DEFCHR$(108)=HEXCHR$("00F800000000000001FBFB010000000000F8F80000000000")
1690 DEFCHR$(109)=HEXCHR$("805F00000000000080DFDF8000000000805F1F0000000000")
1700 DEFCHR$(110)=HEXCHR$("000000000000000000000000000000000000000000000000")
1710 DEFCHR$(111)=HEXCHR$("000000000000000000000000000000000000000000000000")
1720 DEFCHR$(112)=HEXCHR$("00000000000000000000000000000000FEFDFAF5EAD5AA55")
1730 DEFCHR$(113)=HEXCHR$("552A150A05020100552A150A050201007F3F9F4FA753A954")
1740 DEFCHR$(114)=HEXCHR$("000000000000000000000000000000007F7F7F7F7F7F7F7F")
1750 DEFCHR$(115)=HEXCHR$("54AA54AA54AA54AA54AA54AA54AA54AAFEFEFEFEFEFEFEFE")
1760 DEFCHR$(116)=HEXCHR$("00000000000000000000000000000000AA55AA55AA55AA00")
1770 DEFCHR$(117)=HEXCHR$("00000000000000000000000000000000AA55AA55AA55AA00")
1780 DEFCHR$(118)=HEXCHR$("00000000000000000000000000000000FFFFFFFFFFFFFFFF")
1790 DEFCHR$(119)=HEXCHR$("00000000000000000000000000000000FFFFFFFFFFFFFFFF")
1800 DEFCHR$(120)=HEXCHR$("20202000201700006060600060F7F7606060600020170700")
1810 DEFCHR$(121)=HEXCHR$("0202020002E10000060606006EFEF0606060600002E1E000")
1820 DEFCHR$(122)=HEXCHR$("0000002020202020F0600060606060600000000060606060")
1830 DEFCHR$(123)=HEXCHR$("0000002020202020F0600060606060600000000006060606")
1840 DEFCHR$(124)=HEXCHR$("000000000F8000000000000001FBFB010000000000F8F800")
1850 DEFCHR$(125)=HEXCHR$("0000000805F00000000000000080DFDF800000000805F1F00")
1860 DEFCHR$(126)=HEXCHR$("000000000000000000000000000000000000000000000000")
1870 DEFCHR$(127)=HEXCHR$("000000000000000000000000000000000000000000000000")
1880 DEFCHR$(128)=HEXCHR$("0000000000000000F3F7FFFFFFFFFFFF0F30408080808080")
1890 DEFCHR$(129)=HEXCHR$("0000000000000000F0FCFEFFFFFFFFFF00C0201010101010")
1900 DEFCHR$(130)=HEXCHR$("0000000000000000FFFFFFFFFFFFFFFF0000000000000000")
1910 DEFCHR$(131)=HEXCHR$("FFFF000000000000FFFFFFFFFFFFFFFFFFFF000000000000")
1920 DEFCHR$(132)=HEXCHR$("000000000000FFFFFFFFFFFFFFFFFFFF000000000000FFFF")
1930 DEFCHR$(133)=HEXCHR$("C0C0C0C0C0C0C0C0FFFFFFFFFFFFFFFFC0C0C0C0C0C0C0C0")
1940 DEFCHR$(134)=HEXCHR$("0303030303030303FFFFFFFFFFFFFFFF0303030303030303")
1950 DEFCHR$(135)=HEXCHR$("FFFFC0C0C0C0C0C0FFFFFFFFFFFFFFFFFFFFC0C0C0C0C0C0")
1960 DEFCHR$(136)=HEXCHR$("FFFF030303030303FFFFFFFFFFFFFFFFFFFF030303030303")
1970 DEFCHR$(137)=HEXCHR$("C0C0C0C0C0C0FFFFFFFFFFFFFFFFFFFFC0C0C0C0C0C0FFFF")
1980 DEFCHR$(138)=HEXCHR$("030303030303FFFFFFFFFFFFFFFFFFFF030303030303FFFF")
1990 DEFCHR$(139)=HEXCHR$("030F1C306060C0C0030F1F3F7F7FFFFF030F1C306060C0C0")
2000 DEFCHR$(140)=HEXCHR$("C0F0380C06060303C0F0F8FCFEFEFFFFC0F0380C06060303")
2010 DEFCHR$(141)=HEXCHR$("C0C06060301C0F03FFFF7F7F3F1F0F03C0C06060301C0F03")
```

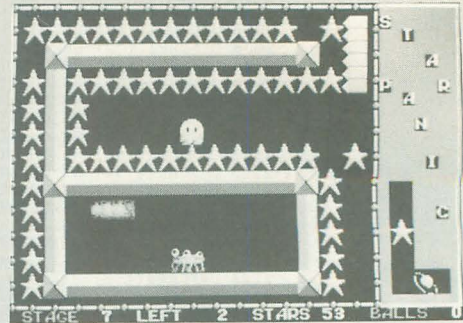
▶私は昔MZユーザーであり,「なぜOh！MZにX1が載るんだ」「パソコン付きテレビX1」などとX1を迫害していました。しかし後になってX1の素晴しさを発見し，そのあと発売されたXlturboに感激して，1985年４月にXlturboユーザーになりました。
泉　昭彦（17）三重県

```
2020 DEFCHR$(142)=HEXCHR$("030306060C38F0C0FFFFFEFEFCF8F0C0030306060C38F0C0")
2030 DEFCHR$(143)=HEXCHR$("000000000000000000000000000000000000000000000000")
2040 DEFCHR$(144)=HEXCHR$("0000000000000000FFFFFFF7F7F3F0F808080804040300F")
2050 DEFCHR$(145)=HEXCHR$("0000000000000000FFFFFFFFEFEFCF0010101010202 0CF0")
2060 DEFCHR$(146)=HEXCHR$("0F3C7060C0C08080FFFCF0E0C0C080800F3C7060C0C08080")
2070 DEFCHR$(147)=HEXCHR$("F03C0E0603030101FF3F0F0703030101F03C0E0603030101")
2080 DEFCHR$(148)=HEXCHR$("8080C0C060703C0F8080C0C0E0F0FCFF8080C0C060703C0F")
2090 DEFCHR$(149)=HEXCHR$("01010303060E3CF0010103030 70F3FFF01010303060E3CF0")
2100 DEFCHR$(150)=HEXCHR$("03060C183060C080FFFEFCF8F0E0C08003060C183060C080")
2110 DEFCHR$(151)=HEXCHR$("80C06030180C060380C0E0F0F8FCFEFF80C06030180C0603")
2120 DEFCHR$(152)=HEXCHR$("0103060C183060C00103070F1F3F7FFF0103060C183060C0")
2130 DEFCHR$(153)=HEXCHR$("C06030180C060301FF7F3F1F0F070301C06030180C060301")
2140 DEFCHR$(154)=HEXCHR$("000000000000000000000000000000000000000000000000")
2150 DEFCHR$(155)=HEXCHR$("000000000000000000000000000000000000000000000000")
2160 DEFCHR$(156)=HEXCHR$("000000000000000000000000000000000000000000000000")
2170 DEFCHR$(157)=HEXCHR$("000000000000000000000000000000000000000000000000")
2180 DEFCHR$(158)=HEXCHR$("000000000000000000000000000000000000000000000000")
2190 DEFCHR$(159)=HEXCHR$("000000000000000000000000000000000000000000000000")
2200 DEFCHR$(160)=HEXCHR$("0101020202048020000000101010 37F1F0000010101037F1F")
2210 DEFCHR$(161)=HEXCHR$("000000000000000038080C0C0C0E0FFFC8080C0C0C0E0FFFC")
2220 DEFCHR$(162)=HEXCHR$("0000010000000000014021100307 0F4F0140211003030703")
2230 DEFCHR$(163)=HEXCHR$("000004084020101000820408C0E0F0F000820408C0E0F0F0")
2240 DEFCHR$(164)=HEXCHR$("000050048 3878F8F00000500483858080000050048 3878F87")
2250 DEFCHR$(165)=HEXCHR$("00000000C0E0F0F000000000C0E0303000000000C0E0F0F0")
2260 DEFCHR$(166)=HEXCHR$("0007180000000000000071F3F6040000000071F3F6040000 0")
2270 DEFCHR$(167)=HEXCHR$("00E0181C0C0E0E0600E0F8FC7C3E3E3E00E0F8FC7C3E3E3E")
2280 DEFCHR$(168)=HEXCHR$("00000001000000000000000100000000000000070F1F1F1F")
2290 DEFCHR$(169)=HEXCHR$("000000E070080808000000E070080808000000E0F0F8F8F8")
2300 DEFCHR$(170)=HEXCHR$("00000070F1F1F1F000000000000000000000000100000000")
2310 DEFCHR$(171)=HEXCHR$("000000E0F0F8F8F8000000000000000000000000E070080808")
2320 DEFCHR$(172)=HEXCHR$("0000000100000000000000070F1F1F1F0000000100000000")
2330 DEFCHR$(173)=HEXCHR$("000000E070080808000000E0F0F8F8F8000000E070080808")
2340 DEFCHR$(174)=HEXCHR$("00000050A150A1500000000 70F1F1F1F00000050A150A15")
2350 DEFCHR$(175)=HEXCHR$("000000E0F058A858000000E0F0F8F8F8000000E0F058A858")
2360 DEFCHR$(176)=HEXCHR$("08080810102020400707070E0C18102007070 70E0C181020")
2370 DEFCHR$(177)=HEXCHR$("0C00804020100804F0F07038180C0402F0F07038180C0402")
2380 DEFCHR$(178)=HEXCHR$("0000000000000000AF0F0703102041008301000000004100")
2390 DEFCHR$(179)=HEXCHR$("0402000000000000F5F2E0C804820080F5F2600804020080")
2400 DEFCHR$(180)=HEXCHR$("4F4723100804020040402010080402004543201008040200")
2410 DEFCHR$(181)=HEXCHR$("F0F0E4C002020250100004000202025 0F0F0E44002020250")
2420 DEFCHR$(182)=HEXCHR$("00000000000000000000040603F1F0700000040603F1F0700")
2430 DEFCHR$(183)=HEXCHR$("06060604000000003E3E7EFCFCF8E0003E3E7EFCFCF8E000")
2440 DEFCHR$(184)=HEXCHR$("00000000000000000000000000000000170F050B01000000")
2450 DEFCHR$(185)=HEXCHR$("0800000000000000080000000000000 0F8F8F8B060000000")
2460 DEFCHR$(186)=HEXCHR$("170F050B010000000000000000000000000000000000000 0")
2470 DEFCHR$(187)=HEXCHR$("F8F8F8B060000000000000000000000008000000000000000")
2480 DEFCHR$(188)=HEXCHR$("0000000000000000170F050B010000000000000000000000")
2490 DEFCHR$(189)=HEXCHR$("0800000000000000F8F8F8B060000000080000000000000 0")
2500 DEFCHR$(190)=HEXCHR$("0205000100000000170F050B010000000205000100000000")
2510 DEFCHR$(191)=HEXCHR$("A850A81000000000F8F8F8B040000000A850A81000000000")
2520 RETURN
```

リスト2　STAR MAIN

```
10 CLEAR &HBA00:LOADM"STAR MON"
20 INIT:CLS4:CGEN1:CLICK OFF
30 FOR C=1 TO 7:COLOR C:FOR I=0 TO 23
40 SCREEN0,1
50 I$=SCRN$(1,I,38)
60 SCREEN0,0
70 LOCATE 1,24:PRINTI$
80 IF STRIG(0) THEN 150
90 NEXT
100 LOCATE 12,24:COLOR 4:CFLASH 1:PRINT"PUSH SPACE KEY";:CFLASH 0
110 FOR I=0 TO 2000
120 IF STRIG(0) THEN 150
130 NEXT:NEXT:GOTO 30
140 '
150 M=0:LT=3:GOSUB 430
160 CLS0
170 LOCATE1,24:COLOR4:PRINT"STAGE    ";
180 COLOR5:PRINT" LEFT  ";:COLOR7:PRINTRIGHT$(STR$(LT),2);
190 COLOR6:PRINT"  STARS    ";
200 COLOR2:PRINT"BALLS";
210 LOCATE 0,0:COLOR 7
220 IF M>29 THEN 280 ELSE POKE &HEF89,M
230 CALL &HED60
240 IF PEEK(&HF8C0)=0 THEN M=M+1:LT=LT+PEEK(&HF8C2):GOTO 160
250 LT=LT-1:IF LT<0 THEN 20
260 IF PEEK(46)=27 THEN M=M+1:IF M>19 THEN M=19
270 GOTO 160
280 INIT:CLS4:CGEN1
290 RESTORE 720:FOR I=1 TO 4
300 READ X,Y,I$
310 COLOR I+2:LOCATE X,Y:'RINT I$;
320 FOR P=1 TO LEN(I$)
330 PP$=MID$(I$,P,1)
340 PRINTPP$;
350 BEEP
360 NEXT
370 NEXT
380 PLAY 300
390 PLAY"O4V15#D3D3E3#E7#D3+#D3B8#B3B3A8G3E3G3G8"
400 READ X,Y,I$:COLOR 7:LOCATE X,Y:CSIZE 1:PRINT#0,I$:COLOR 4,0:CFLASH 1
410 LOCATE12,23:PRINT"PUSH SPACE KEY"
420 IF STRIG(0) THEN 20 ELSE 420
430 COLOR7
```

▶7月号14ページのアートディンクの広告を見て思ったんだけど，女性が着ている服は，
ひょっとしてよくこわれもののの箱に入っているビニール製ショック吸収シートでは？
丸田　明史（16）鳥取県

```
440 RESTORE 470:FOR I=0 TO 23
450 READ I$:LOCATE 32,I:PRINT I$
460 NEXT:RETURN
470 DATA"▂▂▂▂▂▂▂▂▂| "
480 DATA"S▂▂▂▂▂▂▂▂■"
490 DATA"▂▂T▂▂▂▂▂■"
500 DATA"▂▂▂▂▂▂▂▂▂■"
510 DATA"▂▂▂▂▂▂A▂▂■"
520 DATA"▂▂▂▂▂▂▂▂▂■"
530 DATA"P▂▂▂▂▂▂R■"
540 DATA"▂▂A▂▂▂▂▂■"
550 DATA"▂▂▂▂▂▂▂▂▂■"
560 DATA"▂▂▂▂N▂▂■"
570 DATA"▂▂▂▂▂▂▂▂▂■"
580 DATA"▂▂▂▂▂▂▂▂▂■"
590 DATA"▂▂▂▂▂▂I▂■"
600 DATA"▂▂▂▂▂▂▂▂▂■"
610 DATA"▂   ▂▂▂▂■"
620 DATA"▂   ▂▂▂▂■"
630 DATA"▂   ▂▂C■"
640 DATA"▂ 。▂▂▂▂■"
650 DATA"▂-ア▂▂▂▂■"
660 DATA"▂   ▂▂▂▂■"
670 DATA"▂   ▂▂▂▂■"
680 DATA"▂  、・ ■"
690 DATA"▂  ｴｵ ■"
700 DATA"▂▂▂▂▂▂▂▂▂| "
710 '
720 DATA 12,0,"YOU ARE GREAT!"
730 DATA 1,3,"AND #STARS# SHINE IN THE SKY AGAIN !"
740 DATA 4,5,"EVERY LIVING THING IS GLAD TO"
750 DATA 10,7,"SEE THE #STARS#."
760 DATA 5,10,"THE END .... GATERING #STARS#"
```


リスト3 STAR MON

```
BA00 02 02 02 02 02 02 21 00 : 2D
BA08 00 02 02 02 02 02 02 10 : 1C
BA10 00 00 00 00 00 00 06 00 : 06
BA18 00 00 00 00 00 0F 10 00 : 1F
BA20 00 00 00 00 00 00 00 00 : 00
BA28 00 00 00 00 0F 10 00 00 : 1F
BA30 00 00 00 02 04 02 00 00 : 08
BA38 00 00 00 0F 10 00 00 00 : 1F
BA40 00 00 00 00 00 00 00 00 : 00
BA48 00 00 0F 10 00 00 00 00 : 1F
BA50 00 00 06 00 00 00 00 00 : 06
BA58 00 0F 10 00 00 00 00 00 : 1F
BA60 00 21 00 00 00 00 00 00 : 21
BA68 0F 10 00 00 00 00 00 02 : 21
BA70 04 02 00 00 00 00 00 0F : 15
BA78 10 00 00 00 00 00 00 00 : 10
--------------------------------
SUM: 25 46 29 25 27 25 39 21 7178

BA80 00 00 00 00 00 00 0F 10 : 1F
BA88 00 00 00 00 00 00 06 00 : 06
BA90 00 00 00 00 00 0F 02 02 : 13
BA98 02 02 02 02 00 00 00 02 : 0A
BAA0 02 02 02 02 02 02 23 00 : 2F
BAA8 02 04 04 04 04 04 04 04 : 1E
BAB0 04 04 04 02 03 00 00 03 : 14
BAB8 00 00 00 00 00 00 00 00 : 00
BAC0 00 00 03 03 00 00 03 00 : 09
BAC8 00 02 04 04 04 04 02 00 : 14
BAD0 00 03 03 06 00 03 00 00 : 0F
BAD8 03 00 00 00 00 03 00 00 : 06
BAE0 03 03 06 00 03 00 00 03 : 12
BAE8 09 02 00 00 03 00 00 03 : 11
BAF0 03 00 00 03 00 00 02 04 : 0C
BAF8 02 00 00 03 00 00 03 03 : 0B
--------------------------------
SUM: 1E 16 1C 1D 13 1F 48 28 870B

BB00 00 00 03 00 00 00 00 00 : 03
BB08 00 00 03 00 00 03 03 00 : 09
BB10 00 02 04 04 04 04 04 04 : 1A
BB18 04 02 00 00 03 03 00 00 : 0C
BB20 00 00 00 06 06 06 00 00 : 12
BB28 00 00 00 03 03 00 00 00 : 06
BB30 00 00 00 00 00 00 00 00 : 00
BB38 00 00 03 02 04 04 04 04 : 15
BB40 04 04 04 04 04 04 04 04 : 20
BB48 04 02 02 00 00 00 00 00 : 08
BB50 11 11 11 00 00 00 00 00 : 33
BB58 02 03 00 06 06 06 00 00 : 17
BB60 00 00 00 06 06 06 00 03 : 15
BB68 03 00 00 00 00 00 00 00 : 03
BB70 00 00 00 00 00 00 03 03 : 06
BB78 00 00 00 00 00 00 00 00 : 00
--------------------------------
SUM: 22 1E 24 1F 24 24 12 12 5F82
```

```
BB80 00 00 00 00 00 03 02 04 : 09
BB88 04 04 04 04 02 01 02 04 : 19
BB90 04 04 04 04 02 00 00 00 : 12
BB98 00 00 00 00 21 00 00 00 : 21
BBA0 00 00 00 00 02 04 04 04 : 0E
BBA8 04 04 02 01 02 04 04 04 : 19
BBB0 04 04 02 03 00 00 00 00 : 0D
BBB8 00 00 00 00 00 00 00 00 : 00
BBC0 00 03 03 00 00 00 00 00 : 06
BBC8 00 00 00 00 00 00 00 00 : 00
BBD0 03 03 00 06 06 06 00 00 : 18
BBD8 00 00 00 06 06 06 00 03 : 15
BBE0 02 00 00 00 00 00 0E 0E : 1E
BBE8 0E 00 00 00 00 00 02 00 : 10
BBF0 02 00 00 00 00 00 02 00 : 04
BBF8 00 00 00 00 02 00 00 01 : 03
--------------------------------
SUM: 25 16 0F 18 37 18 1E 22 4E31

BC00 00 00 02 00 00 07 00 00 : 09
BC08 02 00 00 01 00 00 02 00 : 05
BC10 00 06 00 00 02 00 00 06 : 0E
BC18 00 00 02 00 00 01 00 00 : 03
BC20 02 00 00 07 00 00 02 00 : 0B
BC28 00 01 00 00 02 00 00 06 : 09
BC30 00 00 02 00 00 06 00 00 : 08
BC38 02 00 00 01 00 00 02 00 : 05
BC40 00 21 00 00 02 00 00 01 : 24
BC48 00 00 02 00 00 06 00 00 : 08
BC50 02 00 00 06 00 00 02 00 : 0A
BC58 00 01 00 00 02 00 00 07 : 0A
BC60 00 00 02 00 00 01 00 00 : 03
BC68 02 00 00 06 00 00 02 00 : 0A
BC70 00 06 00 00 02 00 00 01 : 09
BC78 00 00 02 00 00 07 00 00 : 09
--------------------------------
SUM: 0A 2F 0C 15 0A 1C 0A 15 50F7

BC80 02 00 00 01 00 00 02 00 : 05
BC88 00 00 00 00 02 00 00 00 : 02
BC90 00 00 02 00 00 00 01 06 : 09
BC98 00 00 00 00 00 00 00 00 : 00
BCA0 00 06 01 00 00 00 00 00 : 07
BCA8 00 00 00 00 00 00 00 00 : 00
BCB0 00 00 00 00 00 00 01 06 : 07
BCB8 00 00 00 00 00 06 01 00 : 07
BCC0 00 00 00 00 00 00 00 00 : 00
BCC8 00 00 00 00 00 00 00 00 : 00
BCD0 00 00 00 00 00 00 01 06 : 07
BCD8 00 06 01 00 00 00 00 00 : 07
BCE0 00 00 00 00 00 00 00 00 : 00
BCE8 00 00 00 00 00 00 00 00 : 00
BCF0 00 00 00 21 00 00 01 00 : 22
BCF8 00 00 0E 00 00 00 0F 02 : 1F
--------------------------------
SUM: 02 0C 12 22 02 06 16 14 45B9
```

```
BD00 04 04 04 04 04 02 00 00 : 16
BD08 0F 02 10 00 00 0F 03 05 : 38
BD10 05 05 05 05 03 00 00 0F : 26
BD18 03 10 00 00 0F 03 05 05 : 2F
BD20 02 04 04 02 00 00 0F 03 : 1E
BD28 10 00 0E 0E 02 05 05 02 : 3A
BD30 0E 0E 0E 0E 0E 0E 02 10 : 66
BD38 00 06 02 02 02 00 02 02 : 10
BD40 02 02 02 00 02 02 02 06 : 12
BD48 00 02 00 00 02 00 00 00 : 04
BD50 00 00 02 00 00 02 00 00 : 04
BD58 00 02 02 00 00 00 00 00 : 04
BD60 00 00 02 02 00 00 00 00 : 04
BD68 02 00 00 00 00 00 00 00 : 02
BD70 00 00 02 00 00 00 00 02 : 04
BD78 00 00 00 06 00 06 00 00 : 0C
--------------------------------
SUM: 3F 39 45 31 2C 31 22 38 D07A

BD80 00 02 00 00 00 00 02 00 : 04
BD88 00 00 00 00 00 00 00 00 : 00
BD90 02 00 00 00 01 01 01 01 : 06
BD98 00 00 01 00 00 01 01 01 : 04
BDA0 01 00 00 00 00 01 00 00 : 02
BDA8 00 06 00 00 00 01 00 00 : 07
BDB0 00 00 00 01 00 02 10 00 : 13
BDB8 21 00 0F 02 00 01 00 00 : 33
BDC0 00 00 00 00 00 02 02 02 : 06
BDC8 02 02 00 00 00 00 00 00 : 04
BDD0 00 00 00 00 11 11 11 11 : 44
BDD8 11 00 00 00 00 00 06 06 : 1D
BDE0 06 06 06 06 06 06 06 06 : 30
BDE8 06 06 00 06 0F 00 02 04 : 27
BDF0 04 04 04 04 04 04 04 04 : 20
BDF8 04 02 00 0F 06 03 06 06 : 2A
--------------------------------
SUM: 4B 1C 1A 22 31 27 3F 2F C644

BE00 06 06 06 06 06 06 06 06 : 30
BE08 06 06 0F 06 03 06 00 00 : 2A
BE10 00 00 00 00 00 00 00 00 : 00
BE18 00 00 06 03 06 00 00 00 : 0F
BE20 00 21 00 00 00 00 00 00 : 21
BE28 00 06 03 06 06 06 06 06 : 27
BE30 06 06 06 06 06 06 00 06 : 2A
BE38 06 02 04 04 04 04 04 04 : 20
BE40 04 04 04 04 02 06 00 06 : 1E
BE48 03 00 00 00 00 00 31 00 : 34
BE50 00 00 00 03 06 00 06 03 : 12
BE58 00 00 00 00 00 00 00 00 : 00
BE60 00 00 03 06 00 06 03 00 : 12
BE68 00 00 00 00 41 00 00 00 : 41
BE70 00 03 06 00 06 02 04 04 : 19
BE78 04 04 04 04 04 04 04 04 : 20
--------------------------------
SUM: 23 46 39 30 72 2E 52 27 6941
```

▶BASICのプログラムを，PC-98の一太郎で入力し，X68000 のワープロで修整してX-BASICで実行する，ということを行っています。なかなか便利ですよ。

伊藤　誠治 (27) 宮城県

```
BE80 02 06 00 31 00 00 00 00 : 39
BE88 00 00 00 00 00 00 00 00 : 00
BE90 00 00 00 00 00 00 00 00 : 00
BE98 00 00 00 00 00 00 00 00 : 00
BEA0 00 00 00 00 00 01 00 00 : 01
BEA8 00 00 00 00 01 00 00 00 : 01
BEB0 00 00 00 01 09 01 00 00 : 0B
BEB8 00 00 01 07 01 00 00 00 : 09
BEC0 00 00 00 01 00 02 00 00 : 03
BEC8 00 00 01 00 00 00 00 00 : 01
BED0 00 00 00 00 00 02 00 00 : 02
BED8 00 02 00 00 00 00 00 01 : 03
BEE0 00 00 00 00 00 02 00 02 : 04
BEE8 00 00 02 0B 00 01 06 01 : 15
BEF0 00 00 01 00 02 21 02 00 : 26
BEF8 02 00 00 00 00 01 00 00 : 03
---------------------------------
SUM: 04 08 05 45 0D 2B 08 04 CD94

BF00 01 08 01 00 02 02 00 00 : 0E
BF08 00 00 00 00 00 02 02 00 : 04
BF10 01 00 02 00 02 00 00 00 : 05
BF18 00 00 00 00 00 00 00 00 : 00
BF20 00 00 00 00 00 00 00 00 : 00
BF28 01 00 01 00 00 00 00 21 : 23
BF30 00 00 00 00 00 00 00 00 : 00
BF38 00 02 04 04 04 04 04 04 : 1A
BF40 04 04 04 02 00 00 00 00 : 0E
BF48 03 00 00 00 00 00 00 00 : 03
BF50 00 00 03 00 00 00 00 03 : 06
BF58 00 00 01 00 00 00 01 00 : 02
BF60 00 03 00 00 00 00 03 00 : 06
BF68 00 00 09 09 09 00 00 00 : 1B
BF70 03 00 00 00 00 03 44 00 : 4A
BF78 00 09 00 09 00 00 00 03 : 15
---------------------------------
SUM: 0D 1A 19 18 11 0B 4E 2B E3CC

BF80 00 00 00 00 03 00 00 00 : 03
BF88 09 09 09 00 00 00 03 00 : 1E
BF90 00 00 00 02 00 00 01 00 : 03
BF98 00 00 01 00 00 03 00 00 : 04
BFA0 00 00 00 00 00 41 00 01 : 42
BFA8 00 00 00 00 03 00 00 00 : 03
BFB0 00 02 04 04 04 02 00 02 : 12
BFB8 04 04 04 02 00 00 00 00 : 0E
BFC0 00 00 00 00 00 00 00 00 : 00
BFC8 00 00 00 00 00 00 00 00 : 00
BFD0 00 00 00 06 06 00 00 00 : 0C
BFD8 00 00 00 00 00 00 00 00 : 00
BFE0 00 06 02 06 00 00 00 00 : 0E
BFE8 00 00 00 00 00 00 00 00 : 00
BFF0 06 02 06 00 00 21 00 00 : 2F
BFF8 00 00 00 00 02 02 02 02 : 08
---------------------------------
SUM: 13 17 1A 14 12 69 06 05 36E2

C000 02 02 02 02 02 00 00 00 : 0A
C008 00 00 00 00 00 00 06 02 : 08
C010 00 00 00 02 00 00 00 00 : 02
C018 00 00 01 00 00 06 02 00 : 09
C020 00 00 02 00 00 00 00 00 : 02
C028 00 00 00 00 06 02 00 32 : 3A
C030 00 02 00 00 00 00 00 00 : 02
C038 00 00 00 02 06 00 00 00 : 08
C040 02 00 00 00 00 00 00 00 : 02
C048 00 02 06 00 00 00 00 06 : 0E
C050 02 00 00 00 00 00 02 02 : 06
C058 06 00 00 00 00 00 00 06 : 0C
C060 02 02 00 00 00 00 00 00 : 04
C068 00 00 00 00 00 00 00 00 : 00
C070 00 00 00 00 00 0D 00 00 : 0D
C078 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 0E 08 0B 06 0E 15 0A 42 34E5

C080 00 00 00 00 00 00 00 00 : 00
C088 01 01 01 01 01 01 01 00 : 07
C090 01 01 01 01 01 01 01 08 : 0F
C098 00 00 00 00 00 08 00 08 : 10
C0A0 00 00 00 00 00 08 01 01 : 0A
C0A8 01 01 01 01 01 00 01 01 : 07
C0B0 01 01 01 01 01 08 00 00 : 0D
C0B8 00 00 00 08 00 08 00 00 : 10
C0C0 00 00 00 08 01 01 01 01 : 0C
C0C8 01 01 01 00 01 01 01 01 : 07
C0D0 01 01 01 08 00 00 00 00 : 0B
C0D8 00 08 00 08 00 00 00 00 : 10
C0E0 00 08 01 01 01 01 01 01 : 0E
C0E8 01 00 01 01 01 01 01 01 : 07
C0F0 01 08 00 00 00 00 00 08 : 11
C0F8 00 08 00 00 00 00 00 08 : 10
---------------------------------
SUM: 08 26 08 26 08 26 08 26 AF27

C100 01 01 01 01 01 01 01 01 : 08
C108 00 00 00 00 00 00 00 21 : 21
C110 00 00 00 00 00 00 00 02 : 02
C118 02 02 02 02 02 00 41 00 : 4B
C120 02 02 02 02 02 02 00 00 : 0C
C128 07 07 07 07 00 00 00 07 : 23
```

```
C130 07 07 07 00 00 31 02 02 : 4A
C138 02 02 02 02 00 02 02 02 : 0E
C140 02 02 02 31 00 00 06 06 : 43
C148 06 06 06 02 06 06 06 06 : 2C
C150 06 00 00 00 02 02 02 02 : 0E
C158 02 02 01 02 02 02 02 02 : 0F
C160 02 00 00 00 00 00 00 00 : 02
C168 00 21 00 00 00 00 00 00 : 21
C170 00 00 02 02 02 02 02 02 : 0C
C178 00 02 02 02 02 02 02 00 : 0C
---------------------------------
SUM: 27 42 22 47 13 44 5A 41 4304

C180 00 00 09 09 09 09 09 02 : 2F
C188 09 09 09 09 09 00 00 00 : 2D
C190 02 02 02 02 02 02 00 02 : 0E
C198 02 02 02 02 02 00 00 00 : 0A
C1A0 08 08 08 08 00 43 00 08 : 6B
C1A8 08 08 08 00 00 02 02 02 : 1E
C1B0 02 02 02 00 02 00 02 02 : 0C
C1B8 02 02 02 02 00 01 00 00 : 09
C1C0 00 00 00 00 00 00 00 21 : 21
C1C8 00 00 00 00 00 0E 04 04 : 16
C1D0 02 00 00 00 02 04 04 04 : 10
C1D8 02 00 00 0F 05 05 05 03 : 23
C1E0 00 00 00 03 05 05 05 03 : 15
C1E8 00 00 02 04 04 04 02 00 : 10
C1F0 00 00 02 11 11 11 02 00 : 37
C1F8 00 00 00 00 02 00 00 00 : 02
---------------------------------
SUM: 25 21 2E 47 3B 82 23 3F E342

C200 01 00 00 00 00 00 00 06 : 07
C208 06 06 06 03 00 00 41 00 : 56
C210 00 00 00 00 06 02 00 00 : 08
C218 00 06 03 00 0E 0E 0E 02 : 35
C220 02 02 02 00 00 06 09 02 : 17
C228 06 03 02 04 04 04 02 05 : 1E
C230 05 05 02 00 00 02 03 06 : 17
C238 03 02 04 04 04 02 05 05 : 1D
C240 02 03 00 00 06 03 00 11 : 1F
C248 09 00 11 00 09 11 11 00 : 45
C250 11 00 00 00 02 00 00 00 : 13
C258 00 00 00 00 00 00 00 00 : 00
C260 00 06 02 06 02 06 02 06 : 1E
C268 02 02 02 00 06 06 02 23 : 37
C270 06 02 06 00 06 02 06 00 : 1C
C278 00 02 00 02 02 02 00 06 : 0E
---------------------------------
SUM: 3B 27 2E 13 3D 42 7D 5A E284

C280 02 02 02 00 02 02 02 00 : 0C
C288 02 00 02 02 00 00 00 00 : 06
C290 00 00 00 00 41 00 00 00 : 41
C298 02 00 00 00 02 02 00 02 : 08
C2A0 02 02 02 02 02 02 02 02 : 10
C2A8 02 00 02 02 00 00 00 00 : 06
C2B0 02 06 02 06 31 00 02 00 : 43
C2B8 00 00 02 00 02 02 00 02 : 08
C2C0 06 02 06 02 00 02 02 00 : 14
C2C8 02 02 31 02 00 00 02 06 : 3F
C2D0 06 06 02 00 02 00 00 00 : 10
C2D8 02 00 02 00 02 02 02 02 : 0C
C2E0 02 02 00 02 00 02 00 02 : 0A
C2E8 00 02 00 00 00 00 00 00 : 02
C2F0 00 00 00 00 02 00 02 00 : 04
C2F8 00 06 02 00 02 00 00 02 : 0C
---------------------------------
SUM: 1C 20 49 12 82 0E 0E 12 CE40

C300 02 02 06 00 06 02 01 11 : 24
C308 11 11 11 11 11 11 11 11 : 88
C310 11 11 11 11 01 10 00 00 : 55
C318 00 00 00 00 21 00 00 00 : 21
C320 00 00 00 0F 10 00 00 00 : 1F
C328 00 00 06 00 06 00 00 00 : 0C
C330 00 00 0F 10 00 06 00 00 : 25
C338 02 04 04 04 02 00 00 06 : 16
C340 00 0F 10 00 00 00 00 03 : 22
C348 05 05 05 03 00 00 00 00 : 12
C350 0F 10 00 06 00 00 03 05 : 2D
C358 02 05 03 00 00 06 00 0F : 1F
C360 10 00 00 00 00 03 05 05 : 1D
C368 05 03 00 00 00 00 0F 10 : 27
C370 00 00 00 00 02 04 04 04 : 0E
C378 02 00 00 00 00 0F 10 00 : 21
---------------------------------
SUM: 53 54 59 4E 53 45 3D 58 6CEE

C380 02 00 00 00 06 00 06 00 : 0E
C388 00 00 02 00 0F 10 00 00 : 21
C390 00 00 00 00 31 00 00 00 : 31
C398 00 00 00 0F 01 0E 0E 0E : 3A
C3A0 0E 0E 0E 0E 0E 0E 0E 0E : 70
C3A8 0E 0E 01 00 00 00 00 00 : 1D
C3B0 00 02 02 00 02 00 02 00 : 08
C3B8 00 00 00 00 00 00 02 00 : 02
C3C0 00 00 00 02 00 00 00 00 : 02
C3C8 00 00 00 02 00 07 02 00 : 0B
C3D0 08 02 00 00 02 00 00 00 : 0C
C3D8 00 00 02 02 00 00 00 02 : 06
```

```
C3E0 00 07 02 00 00 00 00 00 : 09
C3E8 00 00 01 00 01 00 00 00 : 02
C3F0 01 00 00 02 00 00 00 00 : 03
C3F8 02 07 02 00 01 21 01 00 : 2E
---------------------------------
SUM: 29 2E 1A 27 59 56 27 1E FAF4

C400 00 07 02 00 00 00 00 02 : 0B
C408 00 00 02 00 01 00 02 00 : 05
C410 02 00 00 00 00 00 00 02 : 04
C418 00 07 00 01 00 07 00 00 : 0F
C420 02 00 00 00 00 00 00 02 : 04
C428 02 00 01 00 02 02 00 00 : 07
C430 00 00 00 00 00 00 00 00 : 00
C438 00 01 00 00 00 00 00 00 : 01
C440 00 00 41 00 00 00 00 01 : 42
C448 00 01 00 00 00 00 00 41 : 42
C450 00 00 00 00 00 00 00 00 : 00
C458 00 01 00 00 06 06 06 00 : 13
C460 00 00 00 00 00 00 21 00 : 21
C468 01 00 00 06 06 06 02 02 : 17
C470 02 00 06 02 02 02 02 01 : 11
C478 00 00 02 02 02 02 05 05 : 12
---------------------------------
SUM: 09 11 4E 0B 13 19 32 50 F13E

C480 02 06 02 05 02 00 01 00 : 12
C488 02 05 05 02 00 02 05 05 : 1A
C490 02 00 02 00 00 01 00 00 : 05
C498 02 05 02 00 00 02 05 05 : 15
C4A0 02 00 00 31 01 00 00 02 : 36
C4A8 05 02 00 00 00 02 05 05 : 13
C4B0 02 00 00 01 00 00 02 05 : 0A
C4B8 02 00 00 02 00 02 05 05 : 10
C4C0 02 00 01 00 00 02 05 02 : 0C
C4C8 00 02 05 02 06 02 05 05 : 1B
C4D0 02 01 00 00 02 05 02 02 : 0E
C4D8 02 02 02 06 00 02 02 02 : 12
C4E0 01 00 02 02 02 02 00 00 : 09
C4E8 00 00 00 00 00 00 00 01 : 01
C4F0 00 00 00 00 31 02 04 04 : 3B
C4F8 04 02 00 00 00 02 04 04 : 10
---------------------------------
SUM: 1E 19 15 45 3E 1A 2D 2F CD3B

C500 04 02 00 00 03 00 00 00 : 09
C508 03 00 02 00 03 00 00 00 : 08
C510 03 00 00 03 00 02 00 03 : 0B
C518 00 03 00 03 00 02 00 03 : 0B
C520 00 00 03 00 03 00 03 00 : 09
C528 03 00 03 00 03 00 03 00 : 0C
C530 00 03 00 03 00 03 00 03 : 0C
C538 00 03 00 03 00 03 00 00 : 09
C540 03 00 03 42 03 42 03 42 : D2
C548 03 42 03 00 03 22 00 03 : 70
C550 00 03 00 03 00 03 00 03 : 0C
C558 00 03 00 03 00 00 03 00 : 09
C560 03 00 03 00 03 00 03 00 : 0C
C568 03 00 03 00 00 03 00 03 : 0C
C570 00 03 00 03 00 02 00 03 : 0B
C578 00 03 00 00 02 00 03 00 : 08
---------------------------------
SUM: 19 59 14 57 17 76 12 57 F9B3

C580 02 00 03 00 00 00 03 00 : 08
C588 02 00 0C 07 00 02 00 00 : 17
C590 00 02 04 02 00 02 00 00 : 0A
C598 00 00 00 00 00 00 00 00 : 00
C5A0 00 01 00 00 00 00 00 00 : 01
C5A8 00 06 06 06 02 06 06 06 : 26
C5B0 00 06 06 06 06 06 06 06 : 2A
C5B8 02 02 02 02 02 02 02 00 : 0E
C5C0 02 02 02 02 02 02 02 02 : 10
C5C8 00 00 02 00 00 02 00 02 : 06
C5D0 31 02 00 02 31 02 02 00 : 6A
C5D8 00 02 00 00 02 00 02 02 : 08
C5E0 02 02 02 02 02 02 41 00 : 4D
C5E8 02 41 00 02 21 02 00 00 : 68
C5F0 02 00 00 02 02 02 02 02 : 0C
C5F8 02 02 02 00 02 00 00 02 : 0A
---------------------------------
SUM: 41 5C 29 21 66 1E 5A 16 DD3A

C600 00 00 02 06 06 06 02 06 : 1C
C608 06 06 00 02 00 02 02 02 : 14
C610 00 02 00 00 00 02 00 00 : 04
C618 00 00 02 00 00 00 00 00 : 02
C620 02 00 00 00 02 00 00 00 : 04
C628 00 02 00 00 00 00 00 02 : 04
C630 00 00 00 06 00 00 00 00 : 06
C638 06 00 00 00 00 00 06 11 : 1D
C640 11 11 11 11 11 11 11 11 : 88
C648 11 11 11 11 11 11 00 02 : 68
C650 00 00 00 00 00 00 00 00 : 00
C658 00 00 00 00 00 00 02 00 : 02
C660 02 00 31 00 02 00 00 02 : 37
C668 02 02 00 00 00 02 00 02 : 08
C670 00 00 00 02 00 02 00 00 : 04
C678 02 00 00 00 02 00 00 00 : 06
---------------------------------
SUM: 36 2E 57 32 2E 30 1F 32 1FE5
```

▶遅ればせながらMAGICを入力しました。予想以上に速く描けるので少しグラフィックの世界に入ろうかと思っています。　宮城　裕之（32）神奈川県

```
C680 02 00 02 00 02 00 00 02 : 08
C688 00 00 00 02 00 00 02 00 : 04
C690 02 00 00 00 02 02 00 02 : 08
C698 00 00 09 00 00 09 00 09 : 1B
C6A0 21 00 00 09 00 00 09 00 : 33
C6A8 00 02 00 02 00 00 00 02 : 06
C6B0 00 00 00 00 00 00 00 00 : 00
C6B8 02 02 00 02 00 02 02 02 : 0C
C6C0 00 00 00 00 00 00 00 02 : 02
C6C8 00 00 02 00 00 02 00 00 : 04
C6D0 31 00 00 00 00 0E 02 0E : 4F
C6D8 0E 02 0E 0E 02 02 0E 0E : 4C
C6E0 0E 0E 0E 0E 00 00 00 00 : 38
C6E8 00 00 00 21 00 00 00 00 : 21
C6F0 00 00 00 00 00 02 00 00 : 02
C6F8 00 00 00 00 00 00 00 06 : 06
---------------------------------
SUM: 74 14 29 4C 06 21 1D 35 32E2

C700 00 00 00 02 02 02 0D 02 : 15
C708 0D 0D 02 0D 02 00 02 06 : 33
C710 06 00 00 02 0D 0D 02 0D : 31
C718 0D 02 0D 02 02 00 02 00 : 22
C720 00 00 02 0D 0D 02 0D 02 : 2D
C728 02 0D 02 00 00 06 00 00 : 17
C730 00 02 02 0D 0D 02 0D 02 : 2F
C738 0D 02 00 00 00 31 00 06 : 46
C740 00 00 00 00 00 00 00 00 : 00
C748 00 00 00 00 00 00 00 02 : 02
C750 09 09 09 09 09 02 02 0D : 3E
C758 00 02 02 02 00 00 02 02 : 0A
C760 02 0D 0D 02 0B 0B 02 00 : 36
C768 03 03 03 00 00 02 0C 0C : 23
C770 02 0D 02 0B 0B 02 00 03 : 2C
C778 03 03 31 00 02 02 02 0D : 4A
---------------------------------
SUM: 42 4B 63 45 4E 5D 41 4C 5488

C780 0D 0D 02 02 00 00 02 02 : 22
C788 02 00 00 00 00 00 00 00 : 02
C790 00 00 00 00 00 00 00 00 : 00
C798 00 00 06 00 00 06 00 00 : 0C
C7A0 00 00 00 06 00 00 00 00 : 06
C7A8 00 00 00 00 00 00 00 00 : 00
C7B0 00 00 00 00 00 00 00 06 : 06
C7B8 00 00 00 00 00 06 00 00 : 06
C7C0 00 00 00 09 00 00 00 00 : 09
C7C8 00 00 00 00 00 00 00 00 : 00
C7D0 00 00 00 00 00 00 06 00 : 06
C7D8 00 00 00 00 00 06 00 00 : 06
C7E0 00 00 00 41 00 31 00 00 : 72
C7E8 00 00 21 00 00 00 00 00 : 21
C7F0 00 00 01 01 01 01 01 01 : 06
C7F8 01 01 01 01 01 01 01 01 : 08
---------------------------------
SUM: 10 0E 2B 54 02 3F 0A 10 B450

C800 01 11 11 11 11 11 11 11 : 78
C808 11 11 11 11 11 11 11 00 : 77
C810 00 00 00 00 00 00 00 00 : 00
C818 00 00 00 00 00 00 00 0E : 0E
C820 0E 0E 0E 0E 0E 0E 0E 0E : 70
C828 0E 0E 0E 0E 0E 00 00 00 : 46
C830 00 00 00 00 00 00 00 00 : 00
C838 00 00 00 00 00 00 00 00 : 00
C840 00 00 00 02 02 02 02 02 : 0A
C848 02 00 00 00 00 00 00 00 : 02
C850 00 02 00 00 00 00 00 00 : 02
C858 02 00 00 00 01 01 01 02 : 07
C860 01 01 00 00 00 00 00 00 : 02
C868 02 00 10 00 00 06 02 01 : 1B
C870 00 00 00 00 00 00 00 00 : 00
C878 06 10 00 00 06 01 00 00 : 1D
---------------------------------
SUM: 3B 51 4E 40 47 3A 35 32 371D

C880 00 00 00 00 00 00 00 06 : 06
C888 10 00 00 01 02 00 00 00 : 13
C890 21 00 00 00 00 00 06 10 : 37
C898 00 01 00 02 00 00 00 00 : 03
C8A0 00 00 00 00 00 06 00 01 : 07
C8A8 01 01 02 01 01 00 00 00 : 06
C8B0 00 00 00 00 06 00 00 00 : 06
C8B8 00 02 00 00 00 00 00 00 : 02
C8C0 00 02 00 06 00 00 00 00 : 08
C8C8 06 02 02 02 02 02 02 02 : 14
C8D0 00 31 06 09 09 09 09 09 : 64
C8D8 02 00 00 00 00 41 00 00 : 43
C8E0 00 00 02 00 00 02 00 02 : 06
C8E8 02 02 02 02 02 02 02 02 : 10
C8F0 00 02 00 00 02 02 02 00 : 08
C8F8 00 00 00 00 00 00 02 00 : 02
---------------------------------
SUM: 3C 3D 0E 17 18 58 17 26 9552

C900 02 00 00 00 00 01 00 00 : 03
C908 00 02 02 02 00 00 00 02 : 08
C910 02 00 00 00 01 00 00 02 : 05
C918 00 00 00 02 00 00 06 02 : 0A
C920 00 00 00 02 02 00 01 06 : 0B
C928 00 00 00 02 00 00 06 02 : 0A
C930 02 00 00 00 00 01 06 02 : 0B
C938 00 00 02 00 00 00 06 02 : 0A
C940 00 00 02 00 00 02 00 00 : 04
C948 00 01 00 00 00 00 06 02 : 09
C950 00 02 00 00 00 00 00 01 : 03
C958 00 00 00 00 00 00 06 02 : 08
C960 02 02 02 02 02 02 02 00 : 0E
C968 00 00 00 00 00 00 00 00 : 00
C970 00 21 00 00 00 00 00 00 : 21
C978 00 00 00 00 21 00 00 00 : 21
---------------------------------
SUM: 08 28 08 0A 26 06 27 17 203C

C980 11 11 11 00 00 00 00 00 : 33
C988 00 02 02 02 02 09 00 00 : 11
C990 01 00 00 0A 02 00 00 02 : 0F
C998 00 00 00 00 02 09 00 00 : 0B
C9A0 00 09 02 00 00 00 03 00 : 0E
C9A8 00 00 00 00 02 09 00 09 : 14
C9B0 02 00 00 00 00 03 00 00 : 05
C9B8 00 00 00 00 02 00 02 00 : 04
C9C0 00 00 00 00 02 02 00 00 : 04
C9C8 31 00 00 00 07 00 00 00 : 38
C9D0 31 00 00 00 09 02 00 00 : 3C
C9D8 00 00 02 00 02 00 00 00 : 04
C9E0 00 00 09 02 00 00 00 00 : 0B
C9E8 02 00 00 00 02 00 00 00 : 04
C9F0 00 02 00 00 00 00 02 06 : 0A
C9F8 00 0E 00 06 02 00 00 00 : 16
---------------------------------
SUM: 78 2C 20 14 22 22 07 11 CE0E

CA00 02 04 04 04 04 04 04 04 : 1E
CA08 04 04 04 04 02 00 00 00 : 12
CA10 00 00 00 00 00 00 00 00 : 00
CA18 00 00 00 00 00 00 00 01 : 01
CA20 01 01 01 01 01 01 01 01 : 08
CA28 01 00 00 00 00 00 01 06 : 08
CA30 06 06 06 06 06 06 06 06 : 30
CA38 01 00 00 00 01 06 06 06 : 14
CA40 06 06 06 06 06 06 06 06 : 30
CA48 01 00 01 06 02 04 04 04 : 16
CA50 04 04 04 04 04 04 02 06 : 20
CA58 01 00 00 03 00 00 00 00 : 04
CA60 00 00 00 00 00 03 00 00 : 03
CA68 00 00 03 00 00 06 00 21 : 2A
CA70 00 06 00 00 03 00 00 00 : 09
CA78 00 03 00 02 04 04 04 04 : 15
---------------------------------
SUM: 1B 22 1D 24 21 2C 22 4D 6D22

CA80 04 02 00 03 00 00 00 00 : 09
CA88 03 00 02 04 04 04 04 04 : 19
CA90 02 00 03 00 00 00 00 03 : 08
CA98 00 00 00 00 06 00 00 00 : 06
CAA0 00 03 00 00 00 00 03 00 : 06
CAA8 00 00 00 00 00 00 00 00 : 00
CAB0 02 00 00 00 00 02 00 00 : 04
CAB8 00 00 31 00 00 00 00 01 : 32
CAC0 00 00 31 00 00 00 00 00 : 31
CAC8 00 00 00 00 00 00 00 00 : 00
CAD0 31 00 00 06 00 00 06 00 : 3D
CAD8 00 06 00 00 06 00 00 00 : 0C
CAE0 00 00 00 00 00 00 00 00 : 00
CAE8 00 00 00 00 00 00 00 00 : 00
CAF0 06 00 00 06 00 00 06 00 : 12
CAF8 00 06 00 00 06 00 00 00 : 0C
---------------------------------
SUM: 42 11 67 13 16 06 13 08 D65F

CB00 00 00 00 00 21 01 00 00 : 22
CB08 00 00 00 00 00 00 00 00 : 00
CB10 00 00 00 02 02 00 00 00 : 04
CB18 00 00 00 00 00 06 00 00 : 06
CB20 00 06 00 00 06 00 00 06 : 12
CB28 00 00 00 00 00 00 00 00 : 00
CB30 00 00 00 00 00 00 00 00 : 00
CB38 00 00 06 00 00 06 00 00 : 0C
CB40 06 00 00 06 00 00 06 00 : 12
CB48 00 00 00 00 00 00 00 00 : 00
CB50 00 00 00 00 00 00 00 00 : 00
CB58 31 00 00 00 00 00 00 00 : 31
CB60 00 00 00 00 00 00 31 0C : 3D
CB68 0B 0B 0B 0B 0B 0B 0B 0B : 58
CB70 0B 0B 0B 0B 0B 0B 0C 00 : 4E
CB78 00 02 00 06 00 06 00 00 : 0E
---------------------------------
SUM: 4D 1E 1C 24 3F 23 54 1D 0AB4

CB80 00 00 00 00 0A 0C 02 02 : 1A
CB88 02 02 02 00 00 00 00 00 : 06
CB90 00 00 00 0A 0C 00 00 02 : 18
CB98 00 00 00 00 02 02 02 02 : 08
CBA0 00 00 0A 0C 00 00 02 00 : 18
CBA8 00 00 00 00 00 00 00 00 : 00
CBB0 00 0A 0C 02 00 02 00 02 : 1C
CBB8 00 00 00 00 00 00 00 00 : 00
CBC0 0A 0C 00 00 00 00 00 00 : 16
CBC8 00 00 00 21 00 00 00 0A : 2B
CBD0 0C 00 00 00 02 00 00 00 : 0E
CBD8 00 02 02 02 02 00 0A 0C : 1E
CBE0 00 02 02 02 02 02 00 00 : 0A
CBE8 00 00 00 02 00 0A 0C 00 : 18
CBF0 00 00 02 00 00 00 00 00 : 02
CBF8 00 02 00 00 0A 00 00 00 : 0C
---------------------------------
SUM: 18 1E 1E 3F 26 1E 1C 1E 97EE

CC00 02 00 00 00 00 00 02 02 : 06
CC08 00 02 00 0A 0C 02 04 04 : 22
CC10 04 04 04 04 04 04 04 04 : 20
CC18 04 02 0B 00 00 11 11 11 : 44
CC20 11 11 11 11 11 11 11 11 : 88
CC28 00 00 00 00 00 41 00 00 : 41
CC30 00 00 00 00 00 00 00 00 : 00
CC38 00 00 00 00 02 00 02 00 : 04
CC40 02 00 02 41 00 00 00 00 : 45
CC48 00 00 00 02 00 00 06 00 : 08
CC50 00 02 02 02 00 00 00 00 : 06
CC58 00 00 02 06 06 02 00 00 : 10
CC60 02 00 00 00 00 00 00 00 : 02
CC68 00 02 02 02 02 02 02 02 : 0E
CC70 00 00 00 00 00 02 02 00 : 04
CC78 24 00 00 00 00 00 00 00 : 24
---------------------------------
SUM: 43 1D 28 6C 2B 6F 38 2E 9CB1

CC80 00 00 00 00 00 00 00 0E : 0E
CC88 0E 0E 0E 0E 0E 0E 0E 0E : 70
CC90 0E 00 00 00 02 01 04 04 : 19
CC98 04 04 04 04 04 04 04 04 : 20
CCA0 02 06 00 00 00 00 00 00 : 08
CCA8 00 00 00 00 00 00 00 00 : 00
CCB0 0A 31 00 00 00 00 00 00 : 3B
CCB8 00 00 00 00 00 00 00 31 : 31
CCC0 00 00 00 01 06 01 06 01 : 0F
CCC8 06 01 06 01 00 00 00 00 : 0E
CCD0 00 01 06 01 06 01 06 01 : 16
CCD8 06 01 06 01 00 00 00 00 : 0E
CCE0 00 01 06 01 06 01 06 01 : 16
CCE8 06 01 00 00 00 00 00 00 : 07
CCF0 00 01 06 01 06 01 06 01 : 16
CCF8 00 00 00 00 00 00 02 00 : 02
---------------------------------
SUM: 3E 4F 30 18 2C 17 30 59 C3E8

CD00 00 01 06 01 06 01 00 00 : 0F
CD08 02 00 00 00 00 00 02 00 : 04
CD10 00 01 06 01 00 00 02 00 : 0A
CD18 00 00 00 00 01 00 02 00 : 03
CD20 00 01 00 00 02 00 01 00 : 04
CD28 00 00 01 09 01 00 02 00 : 0D
CD30 00 00 02 00 01 09 01 00 : 0D
CD38 00 00 01 09 01 00 00 02 : 0D
CD40 00 00 01 09 01 00 00 00 : 0B
CD48 00 09 01 09 01 00 21 00 : 35
CD50 01 09 01 09 00 00 00 00 : 14
CD58 00 02 02 00 00 00 00 00 : 04
CD60 02 07 07 15 1F 1F 2A 3F : CC
CD68 3F 15 7F 7F 2A 7F 7F 11 : 8B
CD70 F3 F3 22 F3 F3 11 F3 F3 : E5
CD78 A0 E0 E0 50 F8 F8 A8 FC : 44
---------------------------------
SUM: D7 06 9D 06 42 B1 6F 41 D488

CD80 FC 74 FE FE BA FE FE 4D : 6F
CD88 CF CF 8E CF CF 4F CF CF : B7
CD90 2A FF FF 35 3F FF 2A 3F : 04
CD98 FF 35 3F FF EA 3F 3F 01 : DB
CDA0 DF C3 04 3F 05 00 7E 00 : 68
CDA8 AE FF FF 5D F8 FB AD F8 : A1
CDB0 FB 5D F8 FB AF FC FC 07 : F9
CDB8 BF 87 0A 7E 0A 00 7E 00 : 56
CDC0 02 07 07 15 1F 1F 2A 3F : CC
CDC8 3F 15 7F 7F 2A 7F 7F 15 : 8F
CDD0 FF FF 2A FF FF 31 33 F3 : 7D
CDD8 A0 E0 E0 50 F8 F8 A8 FC : 44
CDE0 FC 74 FE FE BA FE FE 5D : 7F
CDE8 FF FF AE FF FF 4D C8 CB : 8A
CDF0 22 33 F3 31 33 F3 EA 3F : C8
CDF8 3F 15 FF FF 2A FF FF 01 : 7B
---------------------------------
SUM: 77 D3 FD 26 BE 86 0E 06 15FC

CE00 DF C3 04 3F 05 00 7E 00 : 68
CE08 8D C8 CB 4D C8 CB AF FC : AB
CE10 FC 5F FF FF AE FF FF 07 : 0C
CE18 BF 87 0A 7E 0A 00 7E 00 : 56
CE20 02 03 03 05 0F 0F 0A 1F : 54
CE28 1F 15 3F 3F 2A 3F 3F 14 : 6E
CE30 5C 5C 08 5C 5C 14 5C 5C : 44
CE38 A0 F0 F0 54 FC FC AA FE : 74
CE40 FE 74 FE FE BB FF FF 5D : 84
CE48 FF FF AE FF FF 5F FF FF : 07
CE50 2A FF FF 15 FF FF 2A FF : 64
CE58 FF 15 FF FF 28 FD FC 14 : 47
CE60 7D 7C 02 06 06 00 07 00 : 0E
CE68 AE 1F 7F 9F 0F 7F CE 87 : CE
CE70 BF 7E C6 C6 0E FE 1E 2E : 21
CE78 FE 2E 14 FC 14 00 78 00 : C8
---------------------------------
SUM: 52 A3 17 75 2E FF 88 B4 A0D9
```

▶今年の阪神は親切だ。多くのファンに勉強時間を与えてくれる。もっとも僕には関係ないが……。
溝渕　裕史（16）兵庫県

```
CE80 02 07 07 15 1F 1F 2A 3F : CC
CE88 3F 15 7F 7F 2A 7F 7F 11 : 8B
CE90 B9 B9 28 B9 B9 11 B9 B9 : 8F
CE98 A0 E0 E0 50 F8 F8 A8 FC : 44
CEA0 FC 74 FE FE BA FE FE 5D : 7F
CEA8 FF FF AE FF FF 5F FF FF : 07
CEB0 2A FF FF 15 FF FF 2A FF : 64
CEB8 FF 15 FF FF 28 FF FF 14 : 4C
CEC0 FC FC 28 7B 78 00 3B 00 : 4E
CEC8 AE 1F 7F 9F 0F 7F 9E 0F : 26
CED0 7F 9F 0F 7F EE 9F 9F 0F : E7
CED8 FF 0F 16 FE 16 00 F8 00 : 30
CEE0 02 07 07 15 1F 1F 2A 3F : CC
CEE8 3F 15 7F 7F 2A FF FF 11 : 8B
CEF0 73 73 22 73 73 11 73 73 : E5
CEF8 80 C0 C0 50 F0 F0 A8 F8 : D0
----------------------------------
SUM: 1A 54 6C 9C 11 3F E4 4D 5D53

CF00 F8 74 FC FC BA FE FE 5C : 76
CF08 FE FE AE FF FF 5F FF FF : 05
CF10 2A FF FF 15 FE FE 2A FC : 5F
CF18 FD 17 7C 7C 2A 7F 7F 15 : 49
CF20 3F 3F 0A 1E 1E 00 1C 00 : E0
CF28 AE 1F 7F 1F 0F FF 2E 1F : C6
CF30 FF C7 1F 47 82 BE 82 08 : F6
CF38 7C 08 10 F8 10 00 F0 00 : 8C
CF40 02 07 07 15 1F 1F 2A 3F : CC
CF48 3F 15 7F 7F 2A 7F 7F 15 : 8F
CF50 7F 7F 2A FF FF 15 FF FF : 39
CF58 A0 E0 E0 50 F8 F8 A8 FC : 44
CF60 FC 74 FE FE BE FF FF 44 : 6C
CF68 E6 E6 A6 E6 E6 46 E6 E6 : 50
CF70 28 F1 F7 18 F0 F7 2C F8 : 33
CF78 FB 07 DC C4 02 3F 03 05 : EB
----------------------------------
SUM: EA 82 E4 AB 76 BD C6 09 185E

CF80 5F 45 02 0F 02 00 07 00 : BE
CF88 AE FF FF 5F FF FF 2E 7F : B6
CF90 FF DE 7E 7E AE FE FE 5C : DF
CF98 FC FC A8 F8 F8 00 B8 00 : 48
CFA0 02 07 07 15 1F 1F 2A 3F : CC
CFA8 3F 15 7F 7F 2A 7F 7F 15 : 8F
CFB0 FF FF 2A FF FF 15 FF FF : 39
CFB8 A0 E0 E0 50 F8 F8 A8 FC : 44
CFC0 FC 74 FE FE BE FE FE 4D : 73
CFC8 CD CD 8C CD CD 4D CD CD : A7
CFD0 2A F1 F7 11 E1 EF 30 E1 : 04
CFD8 EF 11 E1 EF 2E F1 F1 00 : E0
CFE0 EF E0 01 5F 41 00 1F 00 : 8F
CFE8 AE FF FF 5F FF FF AE FF : B6
CFF0 FF 5F FF FF AE FF FF 5F : 67
CFF8 FF FF 2E FE 3E 00 DC 00 : 44
----------------------------------
SUM: 65 99 46 4D AD D1 CF 83 E9FB

D000 02 07 07 15 1F 1F 2A 3F : CC
D008 3F 15 7F 7F 2A 7F 7F 15 : 8F
D010 FF FF 2A FF FF 15 FF FF : 39
D018 80 C0 C0 50 F0 F0 A8 F8 : D0
D020 F8 74 FC FC BC FC FC 18 : 30
D028 9A 9A 8A 9A 9A 1A 9A 9A : 40
D030 28 F1 F7 11 E1 EF 22 C3 : D6
D038 DF 3D 43 43 20 77 70 01 : AA
D040 6F 61 02 1F 02 00 1E 00 : 11
D048 AE FF FF 5F FF FF AE FF : B6
D050 FF 5F FF FF 2E FF 7F 5E : 66
D058 FE 7E 28 78 78 00 E0 00 : 74
D060 00 7E 00 04 3F 05 01 DF : A6
D068 C3 EA 3F 3F 35 3F FF 2A : C8
D070 3F FF 35 3F FF 2A FF FF : D9
D078 00 7E 00 0A 7E 0A 07 BF : D6
----------------------------------
SUM: 75 39 CC 4E 27 95 A9 E5 7DB5

D080 87 AF FC FC 5D F8 FB AD : 2B
D088 F8 FB 5D F8 FB AE FF FF : EF
D090 11 F3 F3 22 F3 F3 11 F3 : 03
D098 F3 2A 7F 7F 15 7F 7F 2A : 58
D0A0 3F 3F 15 1F 1F 02 07 07 : E1′
D0A8 4F CF CF 8E CF CF 4D CF : 35
D0B0 CF BA FE FE 74 FE FE A8 : 9D
D0B8 FC FC 50 F8 F8 A0 E0 E0 : 98
D0C0 00 7E 00 04 3F 05 01 DF : A6
D0C8 C3 2A FF FF 15 FF FF EA : E8
D0D0 3F 3F 31 33 F3 22 33 F3 : 1D
D0D8 00 7E 00 0A 7E 0A 07 BF : D6
D0E0 87 AE FF FF 5F FF FF AF : 3F
D0E8 FC FC 4D C8 CB 8D C8 CB : F8
D0F0 31 33 F3 2A FF FF 15 FF : ·93
D0F8 FF 2A 7F 7F 15 7F 7F 2A : 64
----------------------------------
SUM: 91 F7 EB E8 BD C1 51 45 71E8

D100 3F 3F 15 1F 1F 02 07 07 : E1
D108 4D C8 CB AE FF FF 5D FF : E8
D110 FF BA FE FE 74 FE FE A8 : CD
D118 FC FC 50 F8 F8 A0 E0 E0 : 98
D120 00 07 00 02 06 06 14 7D : A6
D128 7C 28 FD FC 15 FF FF 2A : DA
D130 FF FF 15 FF FF 2A FF FF : 39
D138 00 78 00 14 FC 14 2E FE : C8
D140 2E 0E FE 1E 7E C6 C6 CE : 30
D148 87 BF 9F 0F 7F AE 1F 7F : BF
D150 14 5C 5C 08 5C 5C 14 5C : FC
D158 5C 2A 3F 3F 15 3F 3F 0A : A1
D160 1F 1F 05 0F 0F 02 03 03 : 69
D168 5F FF FF AE FF FF 5D FF : 65
D170 FF BB FF FF 74 FE FE AA : D2
D178 FE FE 54 FC FC A0 F0 F0 : C8
----------------------------------
SUM: A2 8D CF 00 8C 90 08 81 EDAA

D180 00 3B 00 28 7B 78 14 FC : 66
D188 FC 28 FF FF 15 FF FF 2A : 5F
D190 FF FF 15 FF FF 2A FF FF : 39
D198 00 F8 00 16 FE 16 0F FF : 30
D1A0 0F EE 9F 9F 9F 0F 7F 9E : 06
D1A8 0F 7F 9F 0F 7F AE 1F 7F : 07
D1B0 11 B9 B9 28 B9 B9 11 B9 : E7
D1B8 B9 2A 7F 7F 15 7F 7F 2A : 1E
D1C0 3F 3F 15 1F 1F 02 07 07 : E1
D1C8 5F FF FF AE FF FF 5D FF : 65
D1D0 FF BA FE FE 74 FE FE A8 : CD
D1D8 FC FC 50 F8 F8 A0 E0 E0 : 98
D1E0 00 1C 00 0A 1E 1E 15 3F : B6
D1E8 3F 2A 7F 7F 17 7C 7C 2A : A0
D1F0 FC FD 15 FE FE 2A FF FF : 32
D1F8 00 F0 00 10 F8 10 08 7C : 8C
----------------------------------
SUM: B7 D1 80 EB 2E 1F 29 96 A73A

D200 08 82 BE 82 C7 1F 47 2E : 25
D208 1F FF 1F 0F FF AE 1F 7F : 97
D210 11 73 73 22 73 73 11 73 : 83
D218 73 2A FF FF 15 7F 7F 2A : D8
D220 3F 3F 15 1F 1F 02 07 07 : E1
D228 5F FF FF AE FF FF 5C FE : 63
D230 FE BA FE FE 74 FC FC A8 : C8
D238 F8 F8 50 F0 F0 80 C0 C0 : 20
D240 00 07 00 02 0F 02 05 5F : 7E
D248 45 02 3F 03 07 DC C4 2C : 5C
D250 F8 FB 18 F0 F7 28 F1 F7 : 02
D258 00 B8 00 A8 F8 F8 5C FC : A8
D260 FC AE FE FE DE 7E 7E 2E : AE
D268 7F FF 5F FF FF AE FF FF : 87
D270 15 FF FF 2A FF FF 15 7F : CF
D278 7F 2A 7F 7F 15 7F 7F 2A : E4
----------------------------------
SUM: 8B A0 E3 B0 C6 E4 3C 0B CF02

D280 3F 3F 15 1F 1F 02 07 07 : E1
D288 46 E6 E6 A6 E6 E6 44 E6 : AE
D290 E6 BE FF FF 74 FE FE A8 : BA
D298 FC FC 50 F8 F8 A0 E0 E0 : 98
D2A0 00 1F 00 01 5F 41 00 EF : AF
D2A8 E0 2E F1 F1 11 E1 EF 30 : 01
D2B0 E1 EF 11 E1 EF 2A F1 F7 : C3
D2B8 00 DC 00 2E FE 3E 5F FF : A4
D2C0 FF AE FF FF 5F FF FF AE : B6
D2C8 FF FF 5F FF FF AE FF FF : 07
D2D0 15 FF FF 2A FF FF 15 FF : 4F
D2D8 FF 2A 7F 7F 15 7F 7F 2A : 64
D2E0 3F 3F 15 1F 1F 02 07 07 : E1
D2E8 4D CD CD 8C CD CD 4D CD : 27
D2F0 CD BE FE FE 74 FE FE A8 : 9F
D2F8 FC FC 50 F8 F8 A0 E0 E0 : 98
----------------------------------
SUM: 8F 93 58 05 98 A8 2C BC FA77

D300 00 1E 00 02 1F 02 01 6F : B1
D308 61 20 77 70 3D 43 43 22 : 4D
D310 C3 DF 11 E1 EF 28 F1 F7 : 93
D318 00 E0 00 28 78 78 5E FE : 54
D320 7E 2E FF 7F 5F FF FF AE : 35
D328 FF FF 5F FF FF AE FF FF : 07
D330 15 FF FF 2A FF FF 15 FF : 4F
D338 FF 2A 7F 7F 15 7F 7F 2A : 64
D340 3F 3F 15 1F 1F 02 07 07 : E1
D348 1A 9A 9A 8A 9A 9A 18 9A : BE
D350 9A BC FC FC 74 FC FC A8 : 62
D358 F8 F8 50 F0 F0 80 C0 C0 : 20
D360 05 07 07 0B 1F 1F 17 3F : B2
D368 3F 2F 7F 7F 58 78 78 B8 : 6C
D370 F8 F8 55 FF FF AA FF FF : EB
D378 7C 84 F4 8E 87 F6 FC 8F : 8A
----------------------------------
SUM: 58 92 2E 4E 4F 5F 8A EA E93F

D380 8C FA FF FA 50 FF F8 A8 : 6E
D388 FF F8 50 FB F8 A8 FC FC : DA
D390 55 FF FF AA FF FF 50 F8 : 43
D398 F8 28 78 78 55 7F 7F 2A : 8D
D3A0 3F 3F 00 1F 1F 00 07 07 : CA
D3A8 54 FE FE A8 FF FC 52 FF : 44
D3B0 FA A8 FF F8 50 FF F8 F8 : D8
D3B8 FB F8 08 85 F4 08 84 F4 : F4
D3C0 05 06 07 0A 1E 1F 17 3E : AE
D3C8 3E 2F 7F 7F 5C 7E 7E BA : 7D
D3D0 FE FE 55 FF FF AA FF FF : F7
D3D8 F4 1C DC 3E 1F DE FC 3F : 62
D3E0 3C FA FF FA 10 3F 38 28 : DE
D3E8 3F 38 50 FB F8 A8 FC FC : 5A
D3F0 55 FF FF AA FF FF 54 FE : 4D
D3F8 FE 2A 7E 7E 55 7F 7F 2B : A2
----------------------------------
SUM: 63 A0 4E 3E F2 B2 2F 3B 871A

D400 3F 3F 00 1E 1F 00 06 07 : C8
D408 54 FE FE A8 FF FC 12 3F : 44
D410 3A 28 3F 38 50 FF F8 E8 : 08
D418 FB F8 20 1D DC 20 1C DC : 24
D420 0D 0F 0F 2B 3F 3F 57 7F : AA
D428 7F 2F 7F 7F 5D FF FF BA : C1
D430 FF FF 55 FF FF AA FF FF : F9
D438 40 E0 E0 FC FC FC FE FE : F0
D440 FE FC CF EC 52 8F EA 94 : 14
D448 0F E4 30 1F D0 E0 3E 30 : 60
D450 50 F8 F8 A8 F8 F8 55 7F : AC
D458 7F 2A 7F 7F 15 3F 3F 08 : 42
D460 18 18 00 07 07 00 00 00 : 3E
D468 50 FD F0 A2 F3 F2 54 FF : 17
D470 FE A8 FC FC 54 FC FC A8 : 92
D478 FC FC 00 FC FC 00 F8 F8 : E0
----------------------------------
SUM: D1 35 82 93 5A 93 83 2A 9E71

D480 05 07 07 0B 1F 1F 17 3F : B2
D488 3F 2F 7F 7F 5D 7F 7F BA : 81
D490 FF FF 55 FF FF AA FF FF : F9
D498 54 FC FC FE FE FE FE FE : 42
D4A0 FE FC FF FC 72 8F FA 88 : 78
D4A8 07 F0 08 07 F0 F8 0F 08 : 05
D4B0 55 FF FF A8 F8 F8 50 F8 : 33
D4B8 F8 2A 7F 7F 55 7F 7F 2A : 9D
D4C0 3F 3F 00 18 18 00 07 07 : BC
D4C8 50 FB F8 A0 FB F8 54 FC : 26
D4D0 FC AA FF FE 54 FF FE AA : 9E
D4D8 FF FE 00 FE FE 00 FC FC : F1
D4E0 01 03 03 0B 0F 0F 17 1F : 66
D4E8 1F 2F 3F 3F 5D 7F 7F 3A : 61
D4F0 7F 7F 55 FF FF AA FF FF : F9
D4F8 50 F0 F0 F8 F8 F8 F0 FC : 04
----------------------------------
SUM: 62 C9 DA A6 F0 6B 45 A5 3BF7

D500 F0 E4 FE E4 42 BF E2 A0 : 39
D508 0F E0 10 07 F0 98 0B 68 : 01
D510 55 FF FF AA FF FF 50 F8 : 43
D518 F8 28 78 78 55 7F 7F 2A : 8D
D520 3F 3F 00 1F 1F 00 08 08 : CC
D528 54 8C AC BA CE CE 54 FF : 35
D530 FE AA FF FE 54 FF FE A8 : 9E
D538 FC FC 00 F8 F8 00 E0 E0 : A8
D540 00 08 08 0B 1F 1F 1F 3F : B7
D548 3F 28 78 78 58 78 78 BA : 59
D550 FF FF 55 FF FF AA FF FF : F9
D558 40 E0 E0 F8 F8 F8 FC FC : E0
D560 FC FE FF FE 54 FF FE AA : F2
D568 FF FE 54 FE FE 9A CF EE : A4
D570 55 FF FF AA FF FF 55 FF : 4F
D578 FF 2A 7F 7F 55 7F 7F 2A : A4
----------------------------------
SUM: A6 90 B6 7B D3 F2 29 6E EE11

D580 3F 3F 00 1F 1F 00 03 03 : C2
D588 14 8F EC 1A 0F EA 34 1F : F5
D590 D4 E0 3E 20 40 FC E0 A0 : CE
D598 E8 E0 00 F4 F4 00 F0 F0 : 90
D5A0 05 07 07 08 18 18 1F 3F : A9
D5A8 3F 2F 7F 7F 58 78 78 B8 : 6C
D5B0 F8 F8 55 FF FF AA FF FF : EB
D5B8 54 FC FC FE FE FE FE FF : 43
D5C0 FE FE FF FE 54 FF FE AA : F4
D5C8 FE FE 54 FF FC A8 FF FC : EE
D5D0 55 FF FF AA FF FF 55 FF : 4F
D5D8 FF 2A 7F 7F 55 7F 7F 2A : A4
D5E0 3F 3F 00 1F 1F 00 07 07 : CA
D5E8 52 FF FA 88 07 F0 08 07 : D9
D5F0 F0 88 07 70 70 8B 88 A8 : 1A
D5F8 FC FC 00 FE FE 00 FC FC : EC
----------------------------------
SUM: 6C 9F D3 0C 07 BE FF 28 9149

D600 00 00 00 03 07 07 18 18 : 41
D608 18 2F 3F 3F 5D 7F 7F 38 : 58
D610 78 78 50 F8 F8 AA FF FF : D8
D618 50 F8 F8 FC FC FC FC FC : 2C
D620 FC FE FE FE 54 FE FE AA : F0
D628 FF FE 54 FF FE A0 FD F0 : DB
D630 55 FF FF AA FF FF 55 FF : 4F
D638 FF 2A 7F 7F 55 7F 7F 2A : A4
D640 3F 3F 00 1F 1F 00 07 07 : CA
D648 54 FE F4 22 3F F2 10 0F : B8
D650 E0 90 07 60 50 8B A8 B8 : 12
D658 CC CC 00 FC FC 00 E0 E0 : 50
D660 3E 21 2F 71 E1 6F 3F F1 : 7F
D668 31 5F FF 5F 0A FF 1F 15 : 2B
D670 FF 1F 0A DF 1F 15 3F 3F : B9
D678 A0 E0 E0 D0 F8 F8 E8 FC : 04
----------------------------------
SUM: 7C DC 6A 78 AA 40 85 FD 7319
```

▶X1turboZを購入しましたが，注文してから品物がくるまで4カ月も待たされました。シャープさんになんとかいってください。気が抜けた。　渡辺　三知雄（32）京都府

```
D680 FC F4 FE FE 1A 1E 1E 1D : 5F
D688 1F 1F AA FF FF 55 FF FF : 39
D690 2A 7F 7F 15 FF 3F 4A FF : C4
D698 5F 15 FF 1F 0A FF 1F 1F : D9
D6A0 DF 1F 10 A1 2F 10 21 2F : 3E
D6A8 AA FF FF 55 FF FF 0A 1F : 24
D6B0 1F 14 1E 1E AA FE FE 54 : 69
D6B8 FC FC 00 F8 F8 00 E0 E0 : A8
D6C0 2F 38 3B 7C F8 7B 3F FC : CC
D6C8 3C 5F FF 5F 08 FC 1C 14 : 2D
D6D0 FC 1C 0A DF 1F 15 3F 3F : B3
D6D8 A0 60 E0 50 78 F8 E8 7C : 04
D6E0 7C F4 FE FE 3A 7E 7E 5D : FF
D6E8 7F 7F AA FF FF 55 FF FF : F9
D6F0 2A 7F 7F 15 FF 3F 48 FC : BF
D6F8 5C 14 FC 1C 0A FF 1F 17 : C7
---------------------------------
SUM: D0 EE 9A 75 CB 53 F5 F6 827B

D700 DF 1F 04 B8 3B 04 38 3B : 6C
D708 AA FF FF 55 FF FF 2A 7F : A4
D710 7F 54 7E 7E AA FE FE D4 : 49
D718 FC FC 00 78 F8 00 60 E0 : A8
D720 02 07 07 3F 3F 3F 7F 7F : CB
D728 7F 3F F3 37 4A F1 57 29 : A3
D730 F0 27 0C F8 0B 07 7C 0C : B5
D738 B0 F0 F0 D4 FC FC EA FE : 44
D740 FE F4 FE FE BA FF FF 5D : 03
D748 FF FF AA FF FF 55 FF FF : F9
D750 0A BF 0F 45 CF 4F 2A FF : 64
D758 7F 15 3F 3F 2A 3F 3F 15 : CF
D760 3F 3F 00 3F 3F 00 1F 1F : 3A
D768 0A 1F 1F 15 1F 1F AA FE : 43
D770 FE 54 FE FE A8 FC FC 10 : FE
D778 18 18 00 E0 E0 00 00 00 : F0
---------------------------------
SUM: 0A 5C 8A F8 04 31 28 BD 98B8

D780 2A 3F 3F 7F 7F 7F 7F 7F : 23
D788 7F 3F FF 3F 4E F1 5F 11 : AB
D790 E0 0F 10 E0 0F 1F F0 10 : 0D
D798 A0 E0 E0 D0 F8 F8 E8 FC : 04
D7A0 FC F4 FE FE BA FE FE 5D : FF
D7A8 FF FF AA FF FF 55 FF FF : F9
D7B0 0A DF 1F 05 DF 1F 2A 3F : 74
D7B8 3F 55 FF 7F 2A FF 7F 55 : 0F
D7C0 FF 7F 00 7F 7F 00 3F 3F : FA
D7C8 AA FF FF 15 1F 1F 0A 1F : 24
D7D0 1F 54 FE FE AA FE FE 54 : 69
D7D8 FC FC 00 18 18 00 E0 E0 : E8
D7E0 0A 0F 0F 1F 1F 1F 0F 3F : D3
D7E8 0F 27 7F 27 42 FD 47 05 : 67
D7F0 F0 07 08 E0 0F 19 D0 16 : ED
D7F8 80 C0 C0 D0 F0 F0 E8 F8 : 90
---------------------------------
SUM: BA 5F 47 8F 56 3A 91 70 E5B1

D800 F8 F4 FC FC BA FE FE 5C : F6
D808 FE FE AA FF FF 55 FF FF : F7
D810 2A 31 35 5D 73 73 2A FF : FC
D818 7F 55 FF 7F 2A FF 7F 15 : 0F
D820 3F 3F 00 1F 1F 00 07 07 : CA
D828 AA FF FF 55 FF FF 0A 1F : 24
D830 1F 14 1E 1E AA FE FE 54 : 69
D838 FC FC 00 F8 F8 00 10 10 : 08
D840 02 07 07 1F 1F 1F 3F 3F : EB
D848 3F 7F FF 7F 2A FF 7F 55 : 39
D850 FF 7F 2A 7F 7F 59 F3 77 : 69
D858 00 10 10 D0 F8 F8 F8 FC : D4
D860 FC 14 1E 1E 1A 1E 1E 5D : FF
D868 FF FF AA FF FF 55 FF FF : F9
D870 28 F1 37 58 F0 57 2C F8 : 13
D878 2B 07 7C 04 02 3F 07 05 : FF
---------------------------------
SUM: 31 E6 B2 C7 E1 3A BE 59 4540

D880 17 07 00 2F 2F 00 0F 0F : 9A
D888 AA FF FF 55 FF FF AA FF : A4
D890 FF 54 FE FE AA FE FE 54 : 49
D898 FC FC 00 F8 F8 00 C0 C0 : 68
D8A0 2A 3F 3F 7F 7F 7F 7F FF : A3
D8A8 7F 7F FF 7F 2A FF 7F 55 : 79
D8B0 7F 7F 2A FF 3F 15 FF 3F : B9
D8B8 A0 E0 E0 10 18 18 F8 FC : 94
D8C0 FC F4 FE FE 1A 1E 1E 1D : 5F
D8C8 1F 1F AA FF FF 55 FF FF : 39
D8D0 4A FF 5F 11 E0 0F 10 E0 : 98
D8D8 0F 11 E0 0E 0E D1 11 15 : 13
D8E0 3F 3F 00 7F 7F 00 3F 3F : FA
D8E8 AA FF FF 55 FF FF AA FF : A4
D8F0 FF 54 FE FE AA FE FE 54 : 49
D8F8 FC FC 00 F8 F8 00 E0 E0 : A8
---------------------------------
SUM: DC 24 29 6D F7 F8 71 34 CAC4

D900 0A 1F 1F 3F 3F 3F 3F 3F : 83
D908 3F 7F 7F 7F 2A 7F 7F 55 : 39
D910 FF 7F 2A FF 7F 05 BF 0F : F9
D918 00 00 00 C0 E0 E0 18 18 : B0
D920 18 F4 FC FC BA FE FE 1C : D6
D928 1E 1E 0A 1F 1F 55 FF FF : D7
D930 2A 7F 2F 44 FC 4F 08 F0 : 5F
D938 07 09 E0 06 0A D1 15 1D : 03
D940 33 33 00 3F 3F 00 07 07 : F2
D948 AA FF FF 55 FF FF AA FF : A4
D950 FF 54 FE FE AA FE FE 54 : 49
D958 FC FC 00 F8 F8 00 E0 E0 : A8
D960 00 00 00 00 00 00 00 00 : 00
D968 00 38 38 38 4C 0C 4C 4C : 98
D970 0C 4C 78 3F 7F 22 09 2B : E4
D978 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 93 BD 8A E3 52 41 93 94 C90B

D980 00 00 00 00 00 00 00 74 : 74
D988 F4 F4 3A BA BA 01 55 55 : 41
D990 1A 25 3F 36 09 2B 46 19 : 47
D998 42 42 00 42 44 00 44 84 : D2
D9A0 00 84 84 00 84 84 00 84 : 94
D9A8 58 A7 FF DC 22 66 C6 18 : 40
D9B0 42 42 00 42 22 00 22 21 : 2B
D9B8 00 21 21 00 21 21 00 21 : A5
D9C0 00 00 00 00 00 00 38 38 : 70
D9C8 38 4C 0C 4C 4C 0C 4C 7C : FC
D9D0 3C 7C 30 17 37 02 29 2B : 8C
D9D8 00 00 00 00 00 00 00 00 : 00
D9E0 00 00 00 00 00 00 00 74 : 74
D9E8 F4 F4 3A BA BA 01 55 55 : 41
D9F0 1A 25 3F 36 09 2B 46 19 : 47
D9F8 42 42 00 42 44 00 44 48 : 96
---------------------------------
SUM: AE 0C D2 E5 7A 71 53 4D F6CD

DA00 00 48 48 00 48 88 00 88 : E8
DA08 58 A7 FF DC 22 66 C6 18 : 40
DA10 42 42 00 42 22 00 22 12 : 1C
DA18 00 12 12 00 12 14 00 14 : 5E
DA20 00 00 00 38 38 38 4C 0C : 00
DA28 4C 4C 0C 4C 7C 3C 7C 38 : 5C
DA30 18 38 10 07 17 02 29 2B : D4
DA38 00 00 00 00 00 00 00 00 : 00
DA40 00 00 00 00 00 00 00 74 : 74
DA48 F4 F4 3A BA BA 01 55 55 : 41
DA50 1A 25 3F 36 09 2B 46 19 : 47
DA58 42 44 00 44 48 00 48 28 : 82
DA60 00 28 28 00 28 50 00 50 : 18
DA68 58 A7 FF DC 22 66 C6 18 : 40
DA70 42 22 00 22 12 00 12 14 : BE
DA78 00 14 14 00 14 0A 00 0A : 50
---------------------------------
SUM: E8 29 29 DB E4 64 94 C5 6CCA

DA80 00 00 00 00 00 00 00 00 : 00
DA88 00 00 00 00 00 00 00 00 : 00
DA90 00 00 00 00 00 00 00 00 : 00
DA98 00 00 00 00 00 00 00 00 : 00
DAA0 00 00 00 00 00 00 00 00 : 00
DAA8 00 00 00 00 00 00 00 00 : 00
DAB0 38 38 38 7C 3C 7C 7C 3C : 94
DAB8 7C 78 17 7F 2A 01 2B 2A : 0A
DAC0 15 3F 56 09 4B 46 19 42 : 9F
DAC8 00 00 00 00 00 00 74 F4 : 68
DAD0 F4 3A BA BA 01 55 55 58 : A5
DAD8 A7 FF DC 22 66 C6 18 42 : 2A
DAE0 00 00 00 00 00 00 00 00 : 00
DAE8 00 00 00 00 00 00 00 2E : 2E
DAF0 2F 2F 5C 5D 5D 80 AA AA : 48
DAF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 93 57 97 3D 75 5E 4B 0E D390

DB00 00 1C 1C 1C 32 12 32 32 : FC
DB08 02 32 1E CE DE 44 90 D4 : A6
DB10 1A E5 FF 3B 44 66 63 18 : 5E
DB18 42 42 00 42 44 00 44 84 : D2
DB20 00 84 84 00 84 84 00 84 : 94
DB28 58 A4 FC 6C 90 D4 62 98 : C2
DB30 42 42 00 42 22 00 22 21 : 2B
DB38 00 21 21 00 21 21 00 21 : A5
DB40 00 00 00 00 00 00 00 00 : 00
DB48 00 00 00 00 00 00 00 2E : 2E
DB50 2F 2F 5C 5D 5D 80 AA AA : 48
DB58 00 00 00 00 00 00 1C 1C : 38
DB60 1C 32 12 32 32 12 32 3E : 46
DB68 0E 3E 0C E4 EC 40 94 D4 : D0
DB70 1A E5 FF 3B 44 66 63 18 : 5E
DB78 42 42 00 42 44 00 44 48 : 96
---------------------------------
SUM: AD C6 53 05 F2 6D 20 66 8033

DB80 00 48 48 00 48 28 00 28 : 28
DB88 58 A4 FC 6C 90 D4 62 98 : C2
DB90 42 42 00 42 22 00 22 12 : 1C
DB98 00 12 12 00 12 11 00 11 : 58
DBA0 00 00 00 00 00 00 00 00 : 00
DBA8 00 00 00 00 00 00 00 2E : 2E
DBB0 2F 2F 5C 5D 5D 80 AA AA : 48
DBB8 00 00 00 1C 1C 1C 32 32 : B8
DBC0 32 32 12 32 3E 0E 3E 1C : 4E
DBC8 04 1C 08 E0 E8 40 94 D4 : 98
DBD0 1A E5 FF 3B 44 66 63 18 : 5E
DBD8 42 44 00 44 48 00 48 28 : 82
DBE0 00 28 28 00 28 50 00 50 : 18
DBE8 58 A4 FC 6C 90 D4 62 98 : C2
DBF0 42 22 00 22 12 00 12 14 : BE
DBF8 00 14 14 00 14 0A 00 0A : 50
---------------------------------
SUM: F5 E8 03 46 15 8B 51 23 FDEA

DC00 00 00 00 00 00 00 00 00 : 00
DC08 00 00 00 00 00 00 00 00 : 00
DC10 00 00 00 00 00 00 00 00 : 00
DC18 00 00 00 00 00 00 00 00 : 00
DC20 00 00 00 00 00 00 00 00 : 00
DC28 00 00 00 00 00 00 00 00 : 00
DC30 00 00 00 00 00 00 2E 2F : 5D
DC38 2F 5C 5D 5D 80 AA AA 1A : 33
DC40 E5 FF 3B 44 66 63 18 42 : 86
DC48 1C 1C 1C 3E 1E 3E 3E 0E : 3A
DC50 3E 1E E6 FE 54 84 D4 54 : 40
DC58 A8 FC 6A 90 D2 62 98 42 : AC
DC60 01 01 01 02 02 02 04 05 : 12
DC68 05 02 02 02 06 07 07 06 : 25
DC70 06 06 04 05 05 00 06 06 : 26
DC78 07 80 87 38 C0 F8 60 80 : DE
---------------------------------
SUM: 29 1A 92 AE F7 32 0B C0 8512

DC80 C0 C0 20 80 C0 20 80 0F : 8F
DC88 C0 CF F0 00 F0 60 80 80 : CF
DC90 00 03 03 01 02 03 0C 0E : 26
DC98 0E 1E 1F 1F 12 12 12 13 : B3
DCA0 12 13 0E 00 0E 00 00 00 : 41
DCA8 00 E0 C0 F0 00 F0 6F 80 : 6F
DCB0 8F 80 60 C0 C0 20 80 40 : CF
DCB8 80 C0 38 00 38 07 00 07 : BE
DCC0 01 01 01 02 02 02 04 05 : 12
DCC8 05 02 02 02 06 07 07 06 : 25
DCD0 06 06 04 05 05 00 06 06 : 26
DCD8 00 80 80 3E C0 FE 61 80 : DD
DCE0 C1 C0 20 80 C7 20 87 08 : 97
DCE8 C0 C8 F0 00 F0 60 80 80 : C8
DCF0 00 03 03 01 02 03 1C 1E : 46
DCF8 1E 3C 3D 3D 26 26 26 26 : 6C
---------------------------------
SUM: 5A 33 6F 55 76 5C C8 D4 5568

DD00 25 27 1C 00 1C 00 00 00 : 84
DD08 00 E0 C0 F0 00 F0 68 80 : 68
DD10 88 87 60 C7 C0 20 80 40 : D6
DD18 80 C0 3E 00 3E 01 00 01 : BE
DD20 01 01 01 02 02 02 04 05 : 12
DD28 05 02 02 02 06 07 07 06 : 25
DD30 06 06 04 05 05 00 06 06 : 26
DD38 00 80 80 39 C0 F9 66 80 : D8
DD40 C6 C1 20 81 CE 20 8E 10 : B4
DD48 C0 D0 E0 00 E0 60 80 80 : B0
DD50 00 03 03 01 02 03 38 3A : 7E
DD58 3A 7C 7D 7D 4E 4C 4E 4C : E4
DD60 49 4D 38 00 38 00 00 00 : 06
DD68 00 E0 C0 E0 00 E0 70 80 : 50
DD70 90 8E 60 CE C1 20 81 46 : F4
DD78 80 C6 39 00 39 00 00 00 : B8
---------------------------------
SUM: 52 68 12 A6 17 E2 E4 2E DCB0

DD80 00 00 00 00 00 00 00 00 : 00
DD88 00 00 00 00 00 00 00 00 : 00
DD90 00 00 00 00 00 00 00 00 : 00
DD98 08 0C 0C 11 16 17 23 2C : AD
DDA0 2E 16 11 14 36 39 3C 30 : 44
DDA8 36 36 27 28 2F 03 34 34 : 55
DDB0 00 00 00 00 00 00 00 00 : 00
DDB8 00 00 00 00 00 00 00 00 : 00
DDC0 00 00 00 00 00 00 00 00 : 00
DDC8 00 1F 1E 0F 10 1F 63 74 : 52
DDD0 74 FC E3 FE F2 F5 F4 FC : 28
DDD8 E0 FC 73 00 73 00 00 00 : C2
DDE0 00 00 00 0E 0E 0E 13 12 : 4F
DDE8 13 12 12 12 1E 19 1F 0C : AB
DDF0 00 0C 01 02 03 00 03 03 : 18
DDF8 07 00 07 38 00 38 40 80 : 3E
---------------------------------
SUM: DA 8D D2 B4 1F C6 5F A1 E1CD

DE00 C0 C0 20 80 80 60 C0 6F : 2F
DE08 80 8F F0 00 F0 00 E0 C0 : 8F
DE10 00 06 06 04 05 05 06 06 : 26
DE18 06 06 07 07 02 02 02 04 : 24
DE20 05 05 02 02 02 01 01 01 : 13
DE28 60 80 80 F0 00 F0 0F C0 : 0F
DE30 CF C0 20 80 C0 20 80 60 : EF
DE38 80 C0 38 C0 F8 07 80 87 : 3E
DE40 00 00 00 1C 1C 1C 26 27 : A1
DE48 27 26 24 26 3C 39 3D 1C : 65
DE50 02 1E 01 02 03 00 03 03 : 2C
DE58 01 00 01 3E 00 3E 40 80 : 3E
DE60 C0 C0 20 80 87 60 C7 68 : 36
DE68 80 88 F0 00 F0 00 E0 C0 : 88
DE70 00 06 06 04 05 05 06 06 : 26
```

▶Woody Pocoが解けた──。もうすぐデーピーソフトから人間になる薬がくるから，それ飲んで人間になるんだーい！

辻村　昌治 (18) 大阪府

```
DE78 06 06 07 07 02 02 02 04 : 24
---------------------------------
SUM: 6A F8 3A CA 0A 79 0D D9 F1CE

DE80 05 05 02 02 02 01 01 01 : 13
DE88 60 80 80 F0 00 F0 08 C0 : 08
DE90 C8 C7 20 87 C0 20 80 61 : F7
DE98 80 C1 3E C0 FE 00 80 80 : 3D
DEA0 00 00 00 38 38 38 4C 4D : 41
DEA8 4D 4E 48 4E 7C 71 7D 38 : D3
DEB0 22 3A 01 02 03 00 03 03 : 68
DEB8 00 00 00 39 00 39 46 80 : 38
DEC0 C6 C1 20 81 8E 60 CE 70 : 54
DEC8 80 90 E0 00 E0 00 E0 C0 : 70
DED0 00 06 06 04 05 05 06 06 : 26
DED8 06 06 07 07 02 02 02 04 : 24
DEE0 05 05 02 02 02 01 01 01 : 13
DEE8 60 80 80 E0 00 E0 10 C0 : F0
DEF0 D0 CE 20 8E C1 20 81 66 : 14
DEF8 80 C6 39 C0 F9 00 80 80 : 38
---------------------------------
SUM: 1D 0B 11 B6 A8 5B E3 8B 05A1

DF00 00 00 00 00 00 00 00 00 : 00
DF08 00 00 00 00 00 00 00 00 : 00
DF10 00 00 00 00 00 00 00 00 : 00
DF18 00 00 00 73 70 73 FC F8 : 4A
DF20 FC F2 E5 F4 FC C3 FE 63 : E7
DF28 14 74 0F 10 1F 00 1F 1E : 03
DF30 00 00 00 00 00 00 00 00 : 00
DF38 00 00 00 00 00 00 00 00 : 00
DF40 00 00 00 00 00 00 00 00 : 00
DF48 03 34 34 27 28 2F 30 36 : 4F
DF50 36 36 39 3C 16 11 14 23 : 3F
DF58 2C 2E 11 16 17 08 0C 0C : B8
DF60 84 00 84 84 00 84 84 00 : 94
DF68 84 44 00 44 42 00 42 46 : D6
DF70 19 42 36 09 2B 1A 25 3F : 43
DF78 21 00 21 21 00 21 21 00 : A5
---------------------------------
SUM: B7 84 4D E2 4D 3D 75 63 117A

DF80 21 22 00 22 42 00 42 C6 : AF
DF88 18 42 DC 22 66 58 A7 FF : BC
DF90 22 09 2B 78 3F 7F 4C 0C : E4
DF98 4C 4C 0C 4C 38 38 38 00 : 98
DFA0 00 00 00 00 00 00 00 00 : 00
DFA8 01 55 55 3A BA BA 74 F4 : C1
DFB0 F4 00 00 00 00 00 00 00 : F4
DFB8 00 00 00 00 00 00 00 00 : 00
DFC0 88 00 88 48 00 48 48 00 : E8
DFC8 48 44 00 44 42 00 42 46 : 9A
DFD0 19 42 36 09 2B 1A 25 3F : 43
DFD8 14 00 14 12 00 12 12 00 : 5E
DFE0 12 22 00 22 42 00 42 C6 : A0
DFE8 18 42 DC 22 66 58 A7 FF : BC
DFF0 02 29 2B 30 17 37 7C 3C : 8C
DFF8 7C 4C 0C 4C 4C 0C 4C 38 : FC
---------------------------------
SUM: 41 6D 4D A9 51 D8 53 83 A997

E000 38 38 00 00 00 00 00 00 : 70
E008 01 55 55 3A BA BA 74 F4 : C1
E010 F4 00 00 00 00 00 00 00 : F4
E018 00 00 00 00 00 00 00 00 : 00
E020 50 00 50 28 00 28 28 00 : 18
E028 28 48 00 48 44 00 44 46 : 86
E030 19 42 36 09 2B 1A 25 3F : 43
E038 0A 00 0A 14 00 14 14 00 : 50
E040 14 12 00 12 22 00 22 C6 : 42
E048 18 42 DC 22 66 58 A7 FF : BC
E050 02 29 2B 10 07 17 38 18 : D4
E058 38 7C 3C 7C 4C 0C 4C 4C : 5C
E060 0C 4C 38 38 38 00 00 00 : 00
E068 01 55 55 3A BA BA 74 F4 : C1
E070 F4 00 00 00 00 00 00 00 : F4
E078 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 2F B1 B5 F9 F6 45 DA 96 E01F

E080 46 19 42 56 09 4B 2A 15 : 8A
E088 3F 2A 01 2B 78 17 7F 7C : 1F
E090 3C 7C 7C 3C 7C 38 38 38 : 94
E098 C6 18 42 DC 22 66 58 A7 : 83
E0A0 FF 01 55 55 3A BA BA 74 : CC
E0A8 F4 F4 00 00 00 00 00 00 : E8
E0B0 00 00 00 00 00 00 00 00 : 00
E0B8 00 00 00 00 00 00 00 00 : 00
E0C0 00 00 00 00 00 00 00 00 : 00
E0C8 00 00 00 00 00 00 00 00 : 00
E0D0 00 00 00 00 00 00 00 00 : 00
E0D8 00 00 00 00 00 00 00 00 : 00
E0E0 84 00 84 84 00 84 84 00 : 94
E0E8 84 44 00 44 42 00 42 63 : F3
E0F0 18 42 3B 44 66 1A E5 FF : 3D
E0F8 21 00 21 21 00 21 21 00 : A5
---------------------------------
SUM; BB 52 36 1B 01 79 BF 46 8381

E100 21 22 00 22 42 00 42 62 : 4B
E108 98 42 6C 90 D4 58 A4 FC : A2
E110 80 AA AA 5C 5D 5D 2E 2F : 47
E118 2F 00 00 00 00 00 00 00 : 2F
E120 00 00 00 00 00 00 00 00 : 00
E128 44 90 D4 1E CE DE 32 02 : A6
E130 32 32 12 32 1C 1C 1C 00 : FC
E138 00 00 00 00 00 00 00 00 : 00
E140 28 00 28 48 00 48 48 00 : 28
E148 48 44 00 44 42 00 42 63 : B7
E150 18 42 3B 44 66 1A E5 FF : 3D
E158 11 00 11 12 00 12 12 00 : 58
E160 12 22 00 22 42 00 42 62 : 3C
E168 98 42 6C 90 D4 58 A4 FC : A2
E170 80 AA AA 5C 5D 5D 2E 2F : 47
E178 2F 00 00 00 00 00 00 00 : 2F
---------------------------------
SUM: D0 64 86 4E 78 D8 F7 7E 1DFA

E180 00 00 00 00 00 00 00 00 : 00
E188 40 94 D4 0C E4 EC 3E 0E : D0
E190 3E 32 12 32 32 12 32 1C : 46
E198 1C 1C 00 00 00 90 00 00 : 38
E1A0 50 00 50 28 00 28 28 00 : 18
E1A8 28 48 00 48 44 00 44 63 : A3
E1B0 18 42 3B 44 66 1A E5 FF : 3D
E1B8 0A 00 0A 14 00 14 14 00 : 50
E1C0 14 12 00 12 22 00 22 62 : DE
E1C8 98 42 6C 90 D4 58 A4 FC : A2
E1D0 80 AA AA 5C 5D 5D 2E 2F : 47
E1D8 2F 00 00 00 00 00 00 00 : 2F
E1E0 00 00 00 00 00 00 00 00 : 00
E1E8 40 94 D4 08 E0 E8 1C 04 : 98
E1F0 1C 3E 0E 3E 32 12 32 32 : 4E
E1F8 32 32 1C 1C 1C 00 00 00 : B8
---------------------------------
SUM: 1D 6E 8F 66 41 03 17 4F E8E6

E200 63 18 42 3B 44 66 1A E5 : A1
E208 FF 80 AA AA 5C 5D 5D 2E : 17
E210 2F 2F 00 00 00 00 00 00 : 5E
E218 62 98 42 6A 90 D2 54 A8 : 04
E220 FC 54 84 D4 1E E6 FE 3E : E8
E228 0E 3E 3E 1E 3E 1C 1C 1C : 3A
E230 00 00 00 00 00 00 00 00 : 00
E238 00 00 00 00 00 00 00 00 : 00
E240 00 00 00 00 00 00 00 00 : 00
E248 00 00 00 00 00 00 00 00 : 00
E250 00 00 00 00 00 00 00 00 : 00
E258 00 00 00 00 00 00 00 00 : 00
E260 E0 00 E0 1C 00 1C 02 01 : FB
E268 03 03 04 01 01 06 03 F6 : 0B
E270 01 F1 0F 00 0F 00 07 03 : 1A
E278 00 00 00 70 70 70 C8 48 : 60
---------------------------------
SUM: E1 E5 E3 CE 0C 29 B9 57 4D9B

E280 C8 48 48 48 78 98 F8 30 : D8
E288 00 30 80 40 C0 00 C0 C0 : 30
E290 06 01 01 0F 00 0F F0 03 : 19
E298 F3 03 04 01 03 04 01 06 : 09
E2A0 01 03 1C 03 1F E0 01 E1 : 04
E2A8 00 60 60 20 A0 A0 60 60 : E0
E2B0 60 60 E0 E0 40 40 40 20 : 60
E2B8 A0 A0 40 40 40 80 80 80 : 80
E2C0 80 00 80 7C 00 7C 02 01 : FB
E2C8 03 03 04 01 E1 06 E3 16 : EB
E2D0 01 11 0F 00 0F 00 07 03 : 3A
E2D8 00 00 00 38 38 38 64 E4 : F0
E2E0 E4 64 24 64 3C 9C BC 38 : 9C
E2E8 40 78 80 40 C0 00 C0 C0 : B8
E2F0 06 01 01 0F 00 0F 10 03 : 39
E2F8 13 E3 04 E1 03 04 01 86 : 69
---------------------------------
SUM: 83 B3 A5 24 A1 54 A7 59 8669

E300 01 83 7C 03 7F 00 01 01 : 84
E308 00 60 60 20 A0 A0 60 60 : E0
E310 60 60 E0 E0 40 40 40 20 : 60
E318 A0 A0 40 40 40 80 80 80 : 80
E320 00 00 00 9C 00 9C 62 01 : 9B
E328 63 83 04 81 71 06 73 0E : 63
E330 01 09 07 00 07 00 07 03 : 22
E338 00 00 00 1C 1C 1C 32 B2 : 38
E340 B2 72 12 72 3E 8E BE 1C : 4E
E348 44 5C 80 40 C0 00 C0 C0 : A0
E350 06 01 01 07 00 07 08 03 : 21
E358 0B 73 04 71 83 04 81 66 : 61
E360 01 63 9C 03 9F 00 01 01 : A4
E368 00 60 60 20 A0 A0 60 60 : E0
E370 60 60 E0 E0 40 40 40 20 : 60
E378 A0 A0 40 40 40 80 80 80 : 80
---------------------------------
SUM: 6D 74 BA E9 73 17 57 0B 41CE

E380 10 30 30 88 68 E8 C4 34 : 40
E388 74 68 88 28 6C 9C 3C 0C : DC
E390 6C 6C E4 14 F4 C0 2C 2C : DC
E398 00 00 00 00 00 00 00 00 : 00
E3A0 00 00 00 00 00 00 00 00 : 00
E3A8 00 00 00 00 00 00 00 00 : 00
E3B0 00 F8 78 F0 08 F8 C6 2E : 54
E3B8 2E 3F C7 7F 4F AF 2F 3F : 1F
E3C0 07 3F CE 00 CE 00 00 00 : E2
E3C8 00 00 00 00 00 00 00 00 : 00
E3D0 00 00 00 00 00 00 00 00 : 00
E3D8 00 00 00 00 00 00 00 00 : 00
E3E0 E0 01 E1 1C 03 1F 06 01 : 07
E3E8 03 03 04 01 03 04 01 F0 : 03
E3F0 03 F3 0F 00 0F 06 01 01 : 1C
E3F8 80 80 80 40 40 40 20 A0 : 00
---------------------------------
SUM: 8B F1 1D 90 42 54 49 6B 86A4

E400 A0 40 40 40 60 E0 E0 60 : E0
E408 60 60 20 A0 A0 00 60 60 : E0
E410 00 07 03 0F 00 0F F6 01 : 1F
E418 F1 01 06 03 03 04 01 02 : 05
E420 01 03 1C 00 1C E0 00 E0 : FC
E428 00 C0 C0 80 40 C0 30 70 : A0
E430 70 78 F8 F8 48 48 48 C8 : 78
E438 48 C8 70 00 70 00 00 00 : F0
E440 00 01 01 7C 03 7F 86 01 : 87
E448 83 03 04 01 E3 04 E1 10 : 63
E450 03 13 0F 00 0F 06 01 01 : 3C
E458 80 80 80 40 40 40 20 A0 : 00
E460 A0 40 40 40 60 E0 E0 60 : E0
E468 60 60 20 A0 A0 00 60 60 : E0
E470 00 07 03 0F 00 0F 16 01 : 3F
E478 11 E1 06 E3 03 04 01 02 : E5
---------------------------------
SUM: C1 CA AA F9 4F 97 8E 50 A94A

E480 01 03 7C 00 7C 80 00 80 : FC
E488 00 C0 C0 80 40 C0 38 78 : B0
E490 78 3C BC BC 64 64 64 64 : BC
E498 A4 E4 38 00 38 00 00 00 : F8
E4A0 00 01 01 9C 03 9F 66 01 : A7
E4A8 63 83 04 81 73 04 71 08 : 5B
E4B0 03 0B 07 00 07 06 01 01 : 24
E4B8 80 80 80 40 40 40 20 A0 : 00
E4C0 A0 40 40 40 60 E0 E0 60 : E0
E4C8 60 60 20 A0 A0 00 60 60 : E0
E4D0 00 07 03 07 00 07 0E 01 : 27
E4D8 09 71 06 73 83 04 81 62 : 5D
E4E0 01 63 9C 00 9C 00 00 00 : 9C
E4E8 00 C0 C0 80 40 C0 1C 5C : 78
E4F0 5C 3E BE BE 72 32 72 32 : 5E
E4F8 92 B2 1C 00 1C 00 00 00 : 7C
---------------------------------
SUM: FB 1D 5B 31 02 6A F1 B7 9EEC

E500 10 30 30 88 68 E8 C4 34 : 40
E508 74 68 88 28 6C 9C 3C 0C : DC
E510 6C 6C E4 14 F4 C0 2C 2C : DC
E518 00 00 00 00 00 00 00 00 : 00
E520 00 00 00 00 00 00 00 00 : 00
E528 00 00 00 00 00 00 00 00 : 00
E530 00 F8 78 F0 08 F8 C6 2E : 54
E538 2E 3F C7 7F 4F AF 2F 3F : 1F
E540 07 3F CE 00 CE 00 00 00 : E2
E548 00 00 00 00 00 00 00 00 : 00
E550 00 00 00 00 00 00 00 00 : 00
E558 00 00 00 00 00 00 00 00 : 00
E560 00 00 00 00 00 00 00 00 : 00
E568 00 0E 00 00 26 00 06 59 : 93
E570 06 1F 36 0F 3F AF 1F 3F : B6
E578 00 00 00 00 00 00 6F 00 : 6F
---------------------------------
SUM: 2B A7 DF 42 52 9A B5 71 DC9E

E580 00 C7 00 00 52 00 52 BF : 2A
E588 00 A4 F5 00 F1 57 80 D6 : 37
E590 AF 1F 3F 76 0F 3F 19 06 : F0
E598 1F 26 00 06 1C 00 00 00 : 67
E5A0 00 00 00 00 00 00 00 00 : 00
E5A8 73 80 F0 EF 00 ED AA 00 : 69
E5B0 AA FF 00 D1 F9 00 00 4D : C0
E5B8 00 00 00 00 00 00 00 00 : 00
E5C0 37 00 02 3D 00 09 37 00 : B6
E5C8 05 24 00 00 0F 00 0B 26 : 69
E5D0 00 06 3B 00 0B 16 01 07 : 6A
E5D8 DC 00 50 F0 00 20 54 00 : 90
E5E0 40 7C 00 60 98 00 90 F8 : 3C
E5E8 00 E0 DC 00 D0 78 80 F0 : 74
E5F0 05 03 07 1B 07 0F 1B 07 : 62
E5F8 0F 15 03 07 06 01 07 0B : 47
---------------------------------
SUM: 57 CD 97 EB F6 4A 5E 0F A7CE

E600 00 03 04 00 00 01 00 00 : 08
E608 A0 C0 E0 D0 E0 F0 D8 E0 : 98
E610 F0 A8 C0 E0 68 80 E0 D0 : D0
E618 00 C0 40 00 00 80 00 00 : 80
E620 00 00 00 00 00 00 B2 00 : B2
E628 00 9F 00 00 FF 00 8B 55 : 7E
E630 00 55 F7 00 B7 CE 01 0F : E1
E638 00 00 00 00 00 00 00 00 : 00
E640 00 38 00 00 64 00 60 98 : 94
E648 60 F8 6E F0 FC F5 F8 FC : 9B
E650 EA 01 6B AF 00 8F FD 00 : 91
E658 25 4A 00 4A E3 00 00 F6 : 92
```

▶昨年11月よりワープロソフトShogunの発売を待ち続けている私です。近日発売予定の近日とは、大目に見ても2，3カ月ではないのか。それを半年以上も待たせるなんて。もう我慢できない。でも，もう少し待とう。
佐藤　裕（39）山梨県

```
E660 00 00 00 00 00 00 00 00 : 00
E668 F5 F8 FC 6C F0 FC 9A 60 : 3B
E670 F8 64 00 60 70 00 00 00 : 2C
E678 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: EC F6 B0 65 A1 3F E5 FE D9A7

E680 01 00 00 02 00 00 0B 00 : 0E
E688 03 16 01 07 15 03 07 1B : 5B
E690 07 0F 0B 07 0F 05 03 07 : 46
E698 80 00 00 20 00 00 D0 00 : 70
E6A0 C0 60 80 E0 A8 C0 E0 D8 : A0
E6A8 E0 F0 D8 E0 F0 A0 C0 E0 : B8
E6B0 1E 01 0F 3B 00 0B 1F 00 : 93
E6B8 07 19 00 09 3E 00 06 2A : 97
E6C0 00 02 0F 00 04 3B 00 0A : 5A
E6C8 68 80 E0 DC 00 D0 64 00 : D8
E6D0 60 F0 00 D0 24 00 00 EC : 30
E6D8 00 A0 BC 00 90 EC 00 40 : 18
E6E0 00 00 00 00 00 00 01 00 : 01
E6E8 00 0E 00 00 26 00 06 59 : 93
E6F0 06 1F B6 0F 3F 2F 1F 3F : B6
E6F8 00 00 00 00 00 00 75 00 : 75
---------------------------------
SUM: 1E CE D4 EF 17 99 A9 D2 900F

E700 00 C7 00 00 5A 00 5A BF : 3A
E708 00 A4 DD 00 D9 5F 80 DE : 17
E710 AF 1F 3F 36 0F 3F 39 06 : D0
E718 1F 26 00 06 1C 00 00 01 : 68
E720 00 00 00 00 00 00 00 00 : 00
E728 6B 80 E8 F7 00 F6 AB 00 : 6B
E730 A9 FF 00 D1 F9 00 00 27 : 99
E738 00 00 00 00 00 00 00 00 : 00
E740 37 00 02 1D 00 09 37 00 : 96
E748 05 0F 00 0B 2F 00 0B 24 : 7D
E750 00 04 3B 00 0B 16 01 07 : 68
E758 FC 00 30 F4 00 40 54 00 : B4
E760 40 B8 00 A0 58 00 50 FC : 3C
E768 00 E0 D8 00 D0 78 80 F0 : 70
E770 25 03 07 1B 07 0F 1B 07 : 82
E778 0F 15 03 07 06 01 07 0B : 47
---------------------------------
SUM: 8E F2 53 E2 C6 7B 47 F4 3908

E780 00 03 04 00 00 02 00 00 : 09
E788 A4 C0 E0 D0 E0 F0 D8 E0 : 9C
E790 F0 A8 C0 E0 68 80 E0 F0 : F0
E798 00 C0 00 00 00 80 00 00 : 40
E7A0 00 00 00 00 00 00 E4 00 : E4
E7A8 00 9F 00 00 FF 00 8B D5 : FE
E7B0 00 95 EF 00 6F D6 01 17 : E1
E7B8 00 00 00 00 00 00 80 00 : 80
E7C0 00 38 00 00 64 00 60 9C : 98
E7C8 60 F8 6C F0 FC F5 F8 FC : 99
E7D0 FA 01 7B BB 00 9B FD 00 : C9
E7D8 25 5A 00 5A E3 00 00 AE : 6A
E7E0 00 00 00 00 00 00 00 00 : 00
E7E8 F4 F8 FC 6D F0 FC 9A 60 : 3B
E7F0 F8 64 00 60 70 00 00 80 : AC
E7F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: FF 46 76 82 59 54 97 E2 6CEA

E800 01 00 00 00 00 00 0F 00 : 10
E808 03 16 01 07 15 03 07 1B : 5B
E810 07 0F 0B 07 0F 25 03 07 : 66
E818 40 00 00 20 00 00 D0 00 : 30
E820 C0 60 80 E0 A8 C0 E0 D8 : A0
E828 E0 F0 D8 E0 F0 A4 C0 E0 : BC
E830 1E 01 0F 1B 00 0B 3F 00 : 93
E838 07 1A 00 0A 1D 00 05 2A : 77
E840 00 02 2F 00 02 3F 00 0C : 7E
E848 68 80 E0 DC 00 D0 24 00 : 98
E850 20 F4 00 D0 F0 00 D0 EC : 90
E858 00 A0 B8 00 90 EC 00 40 : 14
E860 00 00 00 00 00 00 00 00 : 00
E868 00 00 00 00 00 00 00 01 : 01
E870 00 01 03 01 03 07 02 07 : 18
E878 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 98 A7 3D C0 5E 99 C3 44 008D

E880 00 00 00 00 00 00 00 80 : 80
E888 00 80 C0 80 C0 C0 E0 E0 : 00
E890 07 03 07 07 00 07 03 00 : 22
E898 03 00 00 00 00 00 00 00 : 03
E8A0 00 00 00 00 00 00 00 00 : 00
E8A8 E0 C0 E0 C0 00 C0 00 80 : 80
E8B0 80 00 00 00 00 00 00 00 : 80
E8B8 00 00 00 00 00 00 00 00 : 00
E8C0 00 00 00 0A 0F 0F 15 3F : 7C
E8C8 3F 2A 3F 3F 15 7F 7F 2A : 24
E8D0 7F 7F 15 7F 7F 28 79 79 : 2B
E8D8 00 00 00 A0 F0 F0 54 FC : D0
E8E0 FC B8 FC FC 5C FE FE AE : B2
E8E8 FE FE 5C FE FE 8E 9E 9E : 1E
E8F0 11 79 79 2A 7F 7F 15 7F : BF
E8F8 7F 0A 7F 7F 05 3F 3F 02 : 0C
---------------------------------
SUM: B2 25 4B 52 31 77 34 8B 72B3

E900 3F 3F 01 0F 0F 00 00 00 : 9D
E908 1C 9E 9E AA FE FE 5C FE : 58
E910 FE A8 FE FE 54 FC FC A8 : 96
E918 FC FC 50 F0 F0 00 00 00 : 28
E920 00 00 00 00 00 00 05 07 : 0C
E928 07 0A 1F 1F 15 1F 1F 0A : AC
E930 3F 3F 15 3F 3F 08 3D 3D : 93
E938 00 00 00 00 00 00 40 E0 : 20
E940 E0 B8 F8 F8 58 F8 F8 AC : 7C
E948 FC FC 5C FC FC AC BC BC : 70
E950 15 3D 3D 0A 3F 3F 15 3F : 6B
E958 3F 0A 3F 3F 05 1F 1F 00 : 0A
E960 07 07 00 00 00 00 00 00 : 0E
E968 1C BC BC E8 BC FC 5C BC : 4C
E970 FC A8 F8 F8 50 F8 F8 A0 : 74
E978 E0 E0 00 00 00 00 00 00 : C0
---------------------------------
SUM: CA 10 A5 22 49 17 35 D7 06A5

E980 00 00 00 00 00 00 00 00 : 00
E988 00 02 03 03 05 0F 0F 0A : 35
E990 0F 0F 15 1F 1F 08 1D 1D : B3
E998 00 00 00 00 00 00 00 00 : 00
E9A0 00 80 C0 C0 50 F0 F0 A0 : D0
E9A8 F0 F0 58 F8 F8 A8 B8 B8 : 40
E9B0 15 1D 1D 0A 1F 1F 15 1F : CB
E9B8 1F 0A 0F 0F 01 03 03 00 : 4E
E9C0 00 00 00 00 00 00 00 00 : 00
E9C8 18 B8 B8 E8 B8 F8 D0 30 : 20
E9D0 F0 E0 30 F0 40 C0 C0 00 : B0
E9D8 00 00 00 00 00 00 00 00 : 00
E9E0 00 00 00 00 00 00 00 00 : 00
E9E8 00 00 00 00 01 03 03 02 : 09
E9F0 07 07 05 0F 0F 08 0D 0D : 53
E9F8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 42 47 49 DA 94 94 8C DD FF36

EA00 00 00 00 00 40 C0 C0 A0 : 60
EA08 E0 E0 50 F0 F0 A0 B0 B0 : F0
EA10 05 0D 0D 0A 0F 0F 05 07 : 53
EA18 07 02 03 03 00 00 00 00 : 0F
EA20 00 00 00 00 00 00 00 00 : 00
EA28 10 B0 B0 E0 B0 F0 60 20 : 70
EA30 E0 60 20 E0 C0 00 C0 00 : C0
EA38 00 00 00 00 00 00 00 00 : 00
EA40 00 00 00 00 00 00 00 00 : 00
EA48 00 00 00 00 00 00 00 02 : 02
EA50 03 03 05 07 07 00 05 05 : 23
EA58 00 00 00 00 00 00 00 00 : 00
EA60 00 00 00 00 00 00 00 80 : 80
EA68 C0 C0 40 E0 E0 A0 A0 A0 : 60
EA70 05 07 07 02 07 07 01 03 : 27
EA78 03 00 00 00 00 00 00 00 : 03
---------------------------------
SUM: A7 C9 7C A6 9D 06 3B A1 F08E

EA80 00 00 00 00 00 00 00 00 : 00
EA88 40 E0 E0 E0 A0 E0 70 10 : E0
EA90 F0 70 10 F0 70 00 F0 20 : E0
EA98 00 60 00 00 00 00 00 00 : 60
EAA0 00 00 00 00 00 00 00 00 : 00
EAA8 00 00 00 00 00 00 00 00 : 00
EAB0 00 00 01 01 01 00 01 01 : 05
EAB8 00 00 00 00 00 00 00 00 : 00
EAC0 00 00 00 00 00 00 00 00 : 00
EAC8 00 00 00 80 80 E0 80 E0 : 40
EAD0 01 03 03 00 01 01 01 00 : 0A
EAD8 01 01 00 01 01 00 01 00 : 05
EAE0 00 00 00 00 00 00 00 00 : 00
EAE8 70 80 F0 F8 20 F8 F8 10 : F8
EAF0 F8 F8 10 F8 F8 00 F8 F8 : E0
EAF8 00 F8 60 00 60 00 00 00 : B8
---------------------------------
SUM: 9A 24 54 42 0B B9 D3 19 A872

EB00 00 00 00 00 00 00 00 00 : 00
EB08 00 00 00 00 00 00 00 02 : 02
EB10 00 02 00 00 00 05 00 05 : 0C
EB18 00 00 00 00 00 00 00 00 : 00
EB20 00 00 00 00 00 00 00 50 : 50
EB28 00 50 00 00 00 68 00 68 : 20
EB30 08 00 08 01 00 01 0B 00 : 1D
EB38 0B 09 00 09 01 00 01 04 : 23
EB40 00 04 00 00 00 00 00 00 : 04
EB48 F4 00 F4 F8 20 F8 F8 10 : 00
EB50 F8 F8 10 F8 F8 00 F8 F4 : DC
EB58 00 F4 08 00 08 80 00 80 : 04
EB60 00 00 00 10 00 10 04 00 : 24
EB68 04 10 00 10 28 00 28 43 : B7
EB70 00 43 00 00 00 A5 00 A5 : 8D
EB78 40 00 40 20 00 20 82 00 : 42
---------------------------------
SUM: 43 9E 54 3A 49 BB AA 2F 341E

EB80 82 11 00 11 00 00 00 50 : F4
EB88 00 50 A1 00 A1 28 00 28 : E2
EB90 0C 00 0C 41 00 41 0A 00 : A4
EB98 0A 29 00 29 05 00 05 44 : AA
EBA0 00 44 22 00 22 00 00 00 : 88
EBA8 B7 00 B7 DA 00 DA B9 10 : EB
EBB0 B9 E8 00 E8 50 00 50 D5 : FE
EBB8 00 D5 88 00 88 81 00 81 : E7
EBC0 08 00 08 02 00 02 10 00 : 24
EBC8 10 20 00 20 48 00 48 03 : E3
EBD0 00 03 02 00 02 A4 00 A4 : 4F
EBD8 22 00 22 01 00 01 80 00 : C6
EBE0 80 10 00 10 80 00 80 08 : A8
EBE8 00 08 10 00 10 18 00 18 : 58
EBF0 0C 00 0C 81 00 81 12 00 : 2C
EBF8 12 49 00 49 11 00 11 00 : C6
---------------------------------
SUM: E0 0F 56 3A 8B 04 93 E9 486B

EC00 00 00 8C 00 8C 82 00 82 : 1C
EC08 0B 00 0B 09 00 09 09 00 : 31
EC10 09 00 00 00 00 00 00 C3 : CC
EC18 00 C3 94 00 94 89 00 89 : FD
EC20 08 00 08 02 00 02 10 00 : 24
EC28 10 24 00 24 48 00 48 02 : EA
EC30 00 02 00 00 00 A0 00 A0 : 42
EC38 22 00 22 01 00 01 88 00 : CE
EC40 88 00 00 00 00 00 00 00 : 88
EC48 00 00 00 00 00 00 00 00 : 00
EC50 00 00 00 A0 00 A0 00 00 : 40
EC58 00 40 00 40 11 00 11 00 : A2
EC60 00 00 8A 00 8A 02 00 02 : 18
EC68 01 00 01 00 00 00 01 00 : 03
EC70 01 00 00 00 00 00 00 01 : 02
EC78 00 01 94 00 94 88 00 88 : 39
---------------------------------
SUM: D8 2A 74 10 97 E1 FB FB 00B1

EC80 08 00 08 00 00 00 40 00 : 50
EC88 40 00 00 00 40 00 40 00 : C0
EC90 00 00 00 00 00 80 00 80 : 00
EC98 22 00 22 01 00 01 00 00 : 46
ECA0 00 00 00 00 00 00 00 00 : 00
ECA8 00 00 00 00 00 00 00 00 : 00
ECB0 00 00 00 80 00 80 00 00 : 00
ECB8 00 40 00 40 00 00 00 00 : 80
ECC0 00 00 80 00 80 02 00 02 : 04
ECC8 01 00 01 00 00 00 01 00 : 03
ECD0 01 00 00 00 00 00 00 01 : 02
ECD8 00 01 00 00 00 88 00 88 : 11
ECE0 00 00 00 00 00 00 00 00 : 00
ECE8 00 00 00 00 00 00 00 00 : 00
ECF0 00 00 00 00 00 00 00 00 : 00
ECF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 6C 41 AB C1 C0 8B 81 0B CCA3

ED00 00 00 00 00 00 00 00 00 : 00
ED08 00 00 00 00 00 00 00 00 : 00
ED10 00 00 00 00 00 00 00 00 : 00
ED18 00 00 00 00 00 00 00 00 : 00
ED20 00 00 00 00 00 00 00 00 : 00
ED28 00 00 00 00 00 00 00 00 : 00
ED30 00 00 00 00 00 00 00 00 : 00
ED38 00 00 00 00 00 00 00 00 : 00
ED40 00 00 00 00 00 00 00 00 : 00
ED48 00 00 00 00 00 00 00 00 : 00
ED50 00 00 00 00 00 00 00 00 : 00
ED58 00 00 00 00 00 00 00 00 : 00
ED60 AF 32 0E 00 32 0F 00 CD : FD
ED68 B7 ED CD EC ED CD 1A FB : 2C
ED70 C3 8B EF CD 44 FB 2A B9 : 2C
ED78 F8 22 A3 FA 21 02 02 22 : FE
---------------------------------
SUM: 21 CC 6D B3 84 D9 46 A3 14EC

ED80 19 F9 21 C0 E8 06 0B 22 : 0E
ED88 17 F9 CD F7 F8 11 60 00 : 3D
ED90 19 11 00 40 1B 7A B3 20 : D2
ED98 FB 10 EC CD 37 FB 18 03 : 11
EDA0 CD 2A FB 06 32 C5 CD 24 : E0
EDA8 FB 01 00 20 0B 79 B0 20 : 70
EDB0 FB C1 10 F1 C3 44 FB 3E : FD
EDB8 27 32 26 00 11 2B EF CD : 77
EDC0 0B 00 06 0B 11 4C EF CD : 35
EDC8 0B 00 11 5A EF CD 0B 00 : 3D
EDD0 10 F2 11 68 EF 3E 0D CD : 82
EDD8 13 00 CD 0B 00 01 07 18 : 0B
EDE0 ED 43 0E 00 3A 89 EF 3C : 2C
EDE8 CD F0 FA C9 3E 27 32 26 : 3D
EDF0 00 21 B6 F8 11 B7 F8 01 : 90
EDF8 3C 00 36 00 ED B0 ED 4B : 47
---------------------------------
SUM: 5D 77 F4 74 A8 A8 B1 F4 87AA

EE00 89 EF 11 A5 00 CD D5 FA : CA
EE08 11 00 BA 19 E5 DD E1 01 : 88
EE10 01 01 ED 43 0E 00 21 CB : 2C
EE18 F8 11 DB F8 22 F3 F8 ED : D6
EE20 53 F5 F8 16 0B 1E 0F DD : 6B
EE28 7E 00 D5 CD 75 EE D1 87 : DB
EE30 87 01 E3 EE 6F 26 00 09 : F7
EE38 7E CD C8 04 23 7E CD C8 : 4D
EE40 04 ED 4B 0E 00 0D 0D 04 : 68
```

▶三国志の計略でいつも他国の米に火をつけ回っているため放火魔と恐れられている私。将来が心配？
長田　純也（17）岡山県

```
EE48 ED 43 0E 00 23 7E CD C8 : 74
EE50 04 23 7E CD C8 04 3A 0F : 87
EE58 00 3D 32 0F 00 DD 23 1D : 9B
EE60 C2 27 EE 3A 0F 00 3C 3C : 98
EE68 32 0F 00 3E 01 32 0E 00 : C0
EE70 15 C2 25 EE C9 FE 06 28 : DF
EE78 0E FE 07 28 0A FE 08 28 : 73
----------------------------------
SUM: 75 4A 2E 46 F5 E7 0B 6C 6544

EE80 06 FE 09 28 02 18 0A 47 : A0
EE88 3A C1 F8 3C 32 C1 F8 78 : 92
EE90 C9 FE 20 D8 FE 30 DA C8 : 8F
EE98 EE FE 40 DA B6 EE CD D5 : 4C
EEA0 EE EB 2A F5 F8 73 23 72 : F8
EEA8 11 03 00 19 E6 07 77 23 : B4
EEB0 23 22 F5 F8 AF C9 CD D5 : 4C
EEB8 EE EB 2A F3 F8 73 23 72 : F6
EEC0 23 23 23 22 F3 F8 AF C9 : EE
EEC8 CD D5 EE 22 B9 F8 E6 07 : 50
EED0 32 BE F8 AF C9 47 3E 0F : F4
EED8 93 6F 3E 0B 92 67 29 24 : 91
EEE0 2C 78 C9 20 20 20 20 80 : 6D
EEE8 81 90 91 60 61 70 71 62 : A6
EEF0 63 72 73 64 65 74 75 66 : 60
EEF8 67 76 77 A0 A1 B0 B1 A2 : 98
----------------------------------
SUM: 33 CB 35 91 FB FF E6 25 6751

EF00 A3 B2 B3 A4 A5 B4 B5 A6 : 60
EF08 A7 B6 B7 A8 A9 B8 B9 AA : 80
EF10 AB BA BB AC AD BC BD AE : A0
EF18 AF BE BF 08 09 18 19 0A : 78
EF20 0B 1A 1B 0C 0D 1C 1D 0E : A0
EF28 0F 1E 1F 68 6C 6D 6C 6D : 66
EF30 6C 6D 6C 6D 6C 6D 6C 6D : 64
EF38 6C 6D 6C 6D 6C 6D 6C 6D : 64
EF40 6C 6D 6C 6D 6C 6D 6C 6D : 64
EF48 6C 6D 69 00 0D 6A 09 09 : CB
EF50 09 20 20 20 20 20 20 20 : E9
EF58 6B 00 0D 7A 09 09 09 20 : 2D
EF60 20 20 20 20 20 20 7B 00 : 3B
EF68 78 7C 7D 7C 7D 7C 7D 7C : DF
EF70 7D 7C 7D 7C 7D 7C 7D 7C : E4
EF78 7D 7C 7D 7C 7D 7C 7D 7C : E4
----------------------------------
SUM: 74 80 8F E9 8E 37 35 87 B2BC

EF80 7D 7C 7D 7C 7D 7C 7D 79 : E1
EF88 00 00 00 CD D6 EF CD 13 : 72
EF90 F0 CD 48 F1 CD 79 F1 3A : 67
EF98 C1 F8 01 1B 18 ED 43 0E : 2B
EFA0 00 4F CD F0 FA 79 B7 CA : 00
EFA8 A0 ED 01 25 18 ED 43 0E : 09
EFB0 00 3A C2 F8 CD F0 FA 3A : E5
EFB8 C0 F8 B7 C2 73 ED 21 00 : B2
EFC0 30 2B 7C B5 20 FB CD 1B : 8F
EFC8 00 FE 13 CA E8 FA CD 24 : AE
EFD0 FB CD 24 FB 18 B5 3A BB : A9
EFD8 F8 3C E6 03 32 BB F8 AF : B1
EFE0 CD 1B 00 CD 73 F8 21 02 : 43
EFE8 02 22 19 F9 CD 8F F5 2A : B1
EFF0 B9 F8 22 A1 FA ED 4B B6 : 5C
EFF8 F8 7D 81 32 B9 F8 32 A3 : AE
----------------------------------
SUM: 31 93 62 3A CF E5 F2 14 3E28

F000 FA 7C 80 32 BA F8 32 A4 : B0
F008 FA CD 0F F9 CD F7 F8 CD : 58
F010 B3 F0 C9 2A C3 F8 7D B7 : 85
F018 C8 CD 5B F0 D8 2A C3 F8 : 9D
F020 CD 4D 05 44 4D ED 50 CD : BA
F028 32 F1 2A C3 F8 3A C5 F8 : FF
F030 85 6F 3A C6 F8 84 67 CD : A4
F038 4D 05 44 4D ED 51 03 14 : 38
F040 ED 51 21 27 00 09 44 4D : 20
F048 3E 0F 82 ED 79 03 3C ED : 61
F050 79 21 00 00 22 C5 F8 22 : 9B
F058 C3 F8 C9 3A C5 F8 FE 01 : 7A
F060 CA 78 F0 FE FF CA 7C F0 : 65
F068 3A C6 F8 FE 01 CA 98 F0 : 49
F070 FE FF CA 9C F0 C3 67 F5 : 72
F078 2C 2C 18 01 2D CD 4D 05 : BD
----------------------------------
SUM: D5 9A 96 46 C9 FA 27 FD BF39

F080 44 4D ED 78 FE 20 C2 67 : 3D
F088 F5 21 28 00 09 44 4D ED : C5
F090 78 FE 20 C2 67 F5 B7 C9 : 34
F098 24 24 18 01 25 CD 4D 05 : A5
F0A0 44 4D ED 78 FE 20 C2 67 : 3D
F0A8 F5 03 ED 78 FE 20 C2 67 : A4
F0B0 F5 B7 C9 2A B9 F8 CD 4D : 6A
F0B8 05 44 4D ED 78 21 1F F1 : 2C
F0C0 E5 FE A0 CA 13 F1 FE A2 : F1
F0C8 CA 14 F1 FE A4 CA 15 F1 : 41
F0D0 FE A6 CA 16 F1 E1 21 32 : A9
F0D8 F1 E5 FE A8 CA 17 F1 FE : 4C
F0E0 AA CA 17 F1 FE AC CA 17 : 07
F0E8 F1 FE AE CA 17 F1 E1 CD : 1D
F0F0 F9 F0 D0 3E FF 32 C0 F8 : E0
F0F8 C9 FE 20 D8 03 ED 78 FE : 25
----------------------------------
SUM: 03 2E 4B 99 49 EE 8B CB 7A5D

F100 20 D8 21 27 00 09 44 4D : DA
F108 ED 78 FE 20 D8 03 ED 78 : C3
F110 FE 20 C9 C9 C9 C9 C9 3A : 45
F118 C2 F8 3C 32 C2 F8 C9 3A : E5
F120 C1 F8 3D 32 C1 F8 21 00 : 02
F128 00 22 C3 F8 22 C5 F8 CD : 89
F130 67 FC 3E 20 18 01 AF ED : 76
F138 79 03 ED 79 21 27 00 09 : 33
F140 44 4D ED 79 03 ED 79 C9 : 29
F148 21 CB F8 06 04 C5 E5 7E : 16
F150 32 C7 F8 23 7E 32 C8 F8 : 84
F158 23 4E 23 46 23 ED 43 C9 : F6
F160 F8 CD BD F1 E1 ED 5B C7 : 63
F168 F8 73 23 72 23 ED 5B C9 : 34
F170 F8 73 23 72 23 C1 10 D5 : C9
F178 C9 21 DB F8 06 04 C5 E5 : 71
----------------------------------
SUM: D9 82 2D BA 54 22 7F 4E 3441

F180 7E 32 C7 F8 23 7E 32 C8 : 0A
F188 F8 23 4E 23 46 23 ED 43 : 25
F190 C9 F8 7E 32 BF F8 23 7E : C9
F198 32 BD F8 CD B5 F2 E1 ED : 29
F1A0 5B C7 F8 73 23 72 23 ED : 32
F1A8 5B C9 F8 73 23 72 23 3A : 81
F1B0 BF F8 77 23 3A BD F8 77 : B7
F1B8 23 C1 10 C2 C9 3A C7 F8 : 78
F1C0 B7 C8 CD 9A F2 38 0A CD : E7
F1C8 B9 FA 7C AD FE 05 D2 E3 : 94
F1D0 F1 CD 76 F2 22 C9 F8 2A : 33
F1D8 C7 F8 22 A1 FA 22 A3 FA : 3B
F1E0 C3 1C F2 2A C7 F8 E5 CD : 6C
F1E8 4D 05 44 4D CD 32 F1 E1 : B4
F1F0 3A C9 F8 85 6F 3A CA F8 : EB
F1F8 84 67 CD 4D 05 44 4D CD : 68
----------------------------------
SUM: FF 2B DE 08 3A 36 8C 53 1D0A

F200 36 F1 2A C7 F8 22 A1 FA : CD
F208 3A C9 F8 85 32 A3 FA 32 : 81
F210 C7 F8 3A CA F8 84 32 A4 : 15
F218 FA 32 C8 F8 21 02 02 22 : 33
F220 19 F9 CD 2F F2 22 17 F9 : 32
F228 CD 0F F9 CD F7 F8 C9 3A : 94
F230 BC F8 3C E6 01 32 BC F8 : BD
F238 B7 28 05 11 60 E5 18 03 : 55
F240 11 E0 E6 2A C9 F8 7C 87 : C5
F248 85 FE FE CA 71 F2 FE 02 : AE
F250 CA 67 F2 FE FF CA 62 F2 : 3E
F258 FE 01 CA 6C F2 21 80 01 : C9
F260 19 C9 21 00 00 19 C9 21 : 06
F268 60 00 19 C9 21 C0 00 19 : 3C
F270 C9 21 20 01 19 C9 CD B9 : 73
F278 FA 7C AD FE 40 DA 8E F2 : BB
----------------------------------
SUM: 24 B8 D2 27 32 CD 03 81 5BD4

F280 FE 80 DA 92 F2 FE C0 DA : 74
F288 96 F2 21 FF 00 C9 21 01 : 93
F290 00 C9 21 00 FF C9 21 00 : D3
F298 01 C9 2A C7 F8 79 FE 01 : 2B
F2A0 CA 78 F0 FE FF CA 7C F0 : 65
F2A8 78 FE 01 CA 98 F0 FE FF : C6
F2B0 CA 9C F0 37 C9 3A BF F8 : 47
F2B8 B7 C8 ED 4B C7 F8 79 B0 : 9F
F2C0 C8 CD F9 F2 21 02 02 22 : C7
F2C8 19 F9 2A C7 F8 22 A1 FA : B8
F2D0 E5 CD 4D 05 44 4D CD 32 : 94
F2D8 F1 E1 3A C9 F8 85 6F 3A : FB
F2E0 CA F8 84 67 22 A3 FA 22 : 8E
F2E8 C7 F8 CD 4D 05 44 4D CD : 3C
F2F0 36 F1 CD 0F F9 CD F7 F8 : B8
F2F8 C9 3A BF F8 3D CA 0D F3 : C1
----------------------------------
SUM: 9F 6D 9B E4 C2 69 DC D5 CC63

F300 3D CA 35 F3 3D CA 50 F3 : 79
F308 3D CA 6B F3 C9 2A C7 F8 : 17
F310 CD 98 F0 30 13 CD AD F3 : 05
F318 ED 43 C9 F8 CD 9A F2 F5 : 3F
F320 CD FB F3 F1 DC F5 F3 C9 : 39
F328 21 00 01 22 C9 F8 21 E0 : 06
F330 DA 22 17 F9 C9 2A C7 F8 : BE
F338 CD 78 F0 30 56 CD D1 F3 : 4C
F340 ED 43 C9 F8 CD 9A F2 F5 : 3F
F348 CD FB F3 F1 DC F5 F3 C9 : 39
F350 2A C7 F8 CD 9C F0 30 2E : A0
F358 CD AD F3 ED 43 C9 F8 CD : 2B
F360 9A F2 F5 CD FB F3 F1 DC : 09
F368 F5 F3 C9 2A C7 F8 CD 7C : E3
F370 F0 30 2D CD D1 F3 ED 43 : 0E
F378 C9 F8 CD 9A F2 F5 CD FB : D7
----------------------------------
SUM: C2 C3 B3 4B B7 5A E7 B6 A534

F380 F3 F1 DC F5 F3 C9 21 00 : 92
F388 FF 22 C9 F8 21 E0 E0 22 : E5
F390 17 F9 C9 21 01 00 22 C9 : E6
F398 F8 21 E0 DD 22 17 F9 C9 : D1
F3A0 21 FF 00 22 C9 F8 21 E0 : 04
F3A8 E3 22 17 F9 C9 3A C9 F8 : D9
F3B0 B7 FA C3 F3 3A B9 F8 3C : 8E
F3B8 47 3A C7 F8 B8 30 0E 01 : 37
F3C0 01 00 C9 3A B9 F8 47 3A : 36
F3C8 C7 F8 B8 38 F2 01 FF 00 : A1
F3D0 C9 3A CA F8 B7 FA E7 F3 : 50
F3D8 3A BA F8 3C 47 3A C8 F8 : 69
F3E0 B8 30 0E 01 00 01 C9 3A : FB
F3E8 BA F8 47 3A C8 F8 B8 38 : E3
F3F0 F2 01 00 FF C9 21 00 00 : DC
F3F8 22 C9 F8 3A BF F8 21 60 : 55
----------------------------------
SUM: 54 60 7F 0B B4 1A A3 C0 8ABA

F400 D6 11 00 03 19 3D 20 FC : 5C
F408 44 4D 3A BD F8 3C E6 03 : A5
F410 32 BD F8 28 0A 3D 28 07 : 85
F418 11 60 00 19 3D 20 FC ED : D0
F420 5B C9 F8 7B B7 C2 44 F4 : 48
F428 3A BF F8 FE 02 28 05 7A : 98
F430 B7 C2 44 F4 7A ED 44 C2 : 1E
F438 44 F4 60 69 11 80 01 19 : AC
F440 22 17 F9 C9 FA 4B F4 01 : 35
F448 80 01 09 22 17 F9 C9 24 : A9
F450 24 CD 4D 05 44 4D ED 50 : 11
F458 03 ED 58 CD 69 F5 B7 F8 : 22
F460 CB 47 C2 67 F5 CB 5F C2 : 1C
F468 67 F5 CB 42 C2 67 F5 CB : 52
F470 43 CA 67 F5 FE 14 CA 8A : CF
F478 F4 2A B9 F8 24 24 22 C3 : FC
----------------------------------
SUM: 1F BB 1A 2A 33 1D 59 83 D796

F480 F8 21 00 01 22 C5 F8 C3 : BC
F488 67 F5 3A BE F8 CB 4F 28 : 8E
F490 E8 B7 C9 25 CD 4D 05 44 : F0
F498 4D ED 50 03 ED 58 CD 69 : 08
F4A0 F5 B7 F8 CB 47 C2 67 F5 : D4
F4A8 CB 5F C2 67 F5 CB 42 C2 : 17
F4B0 67 F5 CB 43 CA 67 F5 FE : 8E
F4B8 14 CA CD F4 2A B9 F8 25 : 9F
F4C0 25 22 C3 F8 21 00 FF 22 : 44
F4C8 C5 F8 C3 67 F5 3A BE F8 : CC
F4D0 CB 4F 28 E8 B7 C9 2C 2C : 02
F4D8 CD 4D 05 44 4D ED 50 21 : 0E
F4E0 28 00 09 44 4D ED 58 CD : D4
F4E8 69 F5 B7 F8 CB 47 C2 67 : 48
F4F0 F5 CB 5F C2 67 F5 CB 62 : 6A
F4F8 C2 67 F5 CB 63 CA 67 F5 : 72
----------------------------------
SUM: 99 6C 6C A4 00 C5 34 64 0D04

F500 FE 14 CA 16 F5 2A B9 F8 : C2
F508 2C 2C 22 C3 F8 21 01 00 : 57
F510 22 C5 F8 C3 67 F5 3A BE : F6
F518 F8 CB 4F 20 E8 B7 C9 2D : C7
F520 CD 4D 05 44 4D ED 50 21 : 0E
F528 28 00 09 44 4D ED 58 CD : D4
F530 69 F5 B7 F8 CB 47 C2 67 : 48
F538 F5 CB 5F C2 67 F5 CB 62 : 6A
F540 C2 67 F5 CB 63 CA 67 F5 : 72
F548 FE 14 CA 5E F5 2A B9 F8 : 0A
F550 2D 2D 22 C3 F8 21 FF 00 : 57
F558 22 C5 F8 C3 67 F5 3A BE : F6
F560 F8 CB 4F 20 E8 B7 C9 37 : D1
F568 C9 7A CD 77 F5 F8 7B CD : 33
F570 77 F5 BD C8 85 6F C9 FE : AC
F578 21 38 0B FE 80 38 0A FE : 22
----------------------------------
SUM: FF BC 14 0A A1 E4 62 45 ACAA

F580 A0 38 09 3E 14 C9 3E F8 : 32
F588 C9 3E 11 C9 3E 12 C9 3A : 34
F590 BE F8 3D CA A3 F5 3D CA : 5C
F598 57 F6 3D CA 0B F7 3D CA : 5D
F5A0 BF F7 C9 2A B9 F8 CD 4F : 76
F5A8 F4 D2 CE F5 21 60 CD 22 : F9
F5B0 17 F9 3A B7 F8 B7 C2 8B : FD
F5B8 F8 3A B6 F8 FE 01 CA E7 : 90
F5C0 F5 FE FF CA 1F F6 3A B8 : C3
F5C8 F8 B7 C2 DB F5 C9 21 00 : 2B
F5D0 01 22 B6 F8 21 C0 CD 22 : A1
F5D8 17 F9 C9 21 C0 D0 22 17 : C3
F5E0 F9 3E 04 32 BE F8 C9 2A : 16
F5E8 B9 F8 CD D6 F4 DC 8B F8 : A7
F5F0 3A BB F8 87 4F 06 00 21 : EA
F5F8 B1 FA 09 4E 23 46 21 60 : EC
----------------------------------
SUM: E2 1B 2D 04 E9 46 66 3D 0479

F600 CD 09 22 17 F9 2A B9 F8 : E3
F608 2C CD 98 F0 D8 21 01 01 : 7C
F610 22 B6 F8 3E 03 32 BE F8 : F9
F618 21 60 D6 22 17 F9 C9 2A : 7C
F620 B9 F8 CD 1F F5 DC 8B F8 : F1
F628 3A BB F8 87 4F 06 00 21 : EA
```

▶BASICは少し理解できるようになったが、オジンにはたいへんな努力を必要とさせるなあ。ガンバリマス。　細木　稔明（40）富山県

```
F630 A9 FA 09 4E 23 46 21 60 : E4
F638 CD 09 22 17 F9 2A B9 F8 : E3
F640 2D CD 98 F0 D8 21 FF 01 : 7B
F648 22 B6 F8 3E 02 32 BE F8 : F8
F650 21 60 D3 22 17 F9 C9 2A : 79
F658 B9 F8 CD D6 F4 D2 82 F6 : 92
F660 21 60 D3 22 17 F9 3A B6 : 76
F668 F8 B7 C2 8B F8 3A B7 F8 : DD
F670 FE 01 CA 9B F6 FE FF CA : 21
F678 D3 F6 3A B8 F8 B7 C2 8F : BB
--------------------------------
SUM: B8 8B 41 98 2D CE 60 AC 353A

F680 F6 C9 21 01 00 22 B6 F8 : B1
F688 21 C0 D3 22 17 F9 C9 21 : D0
F690 C0 D6 22 17 F9 3E 03 32 : 3B
F698 BE F8 C9 2A B9 F8 CD 4F : 76
F6A0 F4 DC 8B F8 3A BB F8 87 : C7
F6A8 4F 06 00 21 A9 FA 09 4E : 70
F6B0 23 46 21 60 D3 09 22 17 : FF
F6B8 F9 2A B9 F8 24 CD 78 F0 : 2D
F6C0 D8 21 01 01 22 B6 F8 3E : 09
F6C8 04 32 BE F8 21 60 D0 22 : 5F
F6D0 17 F9 C9 2A B9 F8 CD 93 : 14
F6D8 F4 DC 8B F8 3A BB F8 87 : C7
F6E0 4F 06 00 21 B1 FA 09 4E : 78
F6E8 23 46 21 60 D3 09 22 17 : FF
F6F0 F9 2A.B9 F8 25 CD 78 F0 : 2E
F6F8 D8 21 01 FF 22 B6 F8 3E : 07
--------------------------------
SUM: 1E 68 32 68 A4 2B 12 83 2B85

F700 01 32 BE F8 21 60 CD 22 : 59
F708 17 F9 C9 2A B9 F8 CD 1F : A0
F710 F5 D2 36 F7 21 60 D6 22 : 6D
F718 17 F9 3A B6 F8 B7 C2 8B : FC
F720 F8 3A B7 F8 FE FF CA 4F : F7
F728 F7 FE 01 CA 87 F7 3A B8 : 30
F730 F8 B7 C2 43 F7 C9 21 FF : 94
F738 00 22 B6 F8 21 C0 D6 22 : A9
F740 17 F9 C9 21 C0 D3 22 17 : C6
F748 F9 3E 02 32 BE F8 C9 2A : 14
F750 B9 F8 CD 93 F4 DC 8B F8 : 64
F758 3A BB F8 87 4F 06 00 21 : EA
F760 B1 FA 09 4E 23 46 21 60 : EC
F768 D6 09 22 17 F9 2A B9 F8 : EC
F770 25 CD 7C F0 D8 21 FF FF : 55
F778 22 B6 F8 3E 01 32 BE F8 : F7
--------------------------------
SUM: DC 77 56 CC 46 5E 3A BF F176

F780 21 60 CD 22 17 F9 C9 2A : 73
F788 B9 F8 CD 4F F4 DC 8B F8 : 20
F790 3A BB F8 87 4F 06 00 21 : EA
F798 A9 FA 09 4E 23 46 21 60 : E4
F7A0 D6 09 22 17 F9 2A B9 F8 : EC
F7A8 24 CD 7C F0 D8 21 FF 01 : 56
F7B0 22 B6 F8 3E 04 32 BE F8 : FA
F7B8 21 60 D0 22 17 F9 C9 2A : 76
F7C0 B9 F8 CD 93 F4 D2 EA F7 : B8
F7C8 21 60 D0 22 17 F9 3A B7 : 74
F7D0 F8 B7 C2 8B F8 3A B6 F8 : DC
F7D8 FE 01 CA 03 F8 FE FF CA : 8B
F7E0 3B F8 3A B8 F8 B7 C2 F7 : 8D
F7E8 F7 C9 21 00 FF 22 B6 F8 : B0
F7F0 21 C0 D0 22 17 F9 C9 21 : CD
F7F8 C0 CD 22 17 F9 3E 01 32 : 30
--------------------------------
SUM: DD 57 77 E1 6B AA CF 70 FA4A

F800 BE F8 C9 2A B9 F8 CD D6 : FD
F808 F4 DC 8B F8 3A BB F8 87 : C7
F810 4F 06 00 21 B1 FA 09 4E : 78
F818 23 46 21 60 D0 09 22 17 : FC
F820 F9 2A B9 F8 2C CD 9C F0 : 59
F828 D8 21 01 FF 22 B6 F8 3E : 07
F830 03 32 BE F8 21 60 D6 22 : 64
F838 17 F9 C9 2A B9 F8 CD 1F : A0
F840 F5 DC 8B F8 3A BB F8 87 : C8
F848 4F 06 00 21 A9 FA 09 4E : 70
F850 23 46 21 60 D0 09 22 17 : FC
F858 F9 2A B9 F8 2D CD 9C F0 : 5A
F860 D8 21 FF FF 22 B6 F8 3E : 05
F868 02 32 BE F8 21 60 D3 22 : 60
F870 17 F9 C9 FE 32 28 23 FE : 52
F878 34 28 26 FE 36 28 29 FE : 05
--------------------------------
SUM: 94 5C C7 20 27 82 FD 69 C292

F880 38 28 2C FE 20 20 04 3E : 0C
F888 01 18 01 AF 32 B8 F8 21 : CC
F890 00 00 22 B6 F8 B7 C4 5B : A6
F898 FC C9 21 00 01 22 B6 F8 : B7
F8A0 C9 21 FF 00 22 B6 F8 C9 : 82
F8A8 21 01 00 22 B6 F8 C9 21 : DC
F8B0 00 FF 22 B6 F8 C9 00 00 : 98
F8B8 00 00 00 00 00 00 00 00 : 00
F8C0 00 00 00 00 00 00 00 00 : 00
F8C8 00 00 00 00 00 00 00 00 : 00
F8D0 00 00 00 00 00 00 00 00 : 00
```

```
F8D8 00 00 00 00 00 00 00 00 : 00
F8E0 00 00 00 00 00 00 00 00 : 00
F8E8 00 00 00 00 00 00 00 00 : 00
F8F0 00 00 00 00 00 00 00 D9 : D9
F8F8 ED 5B 17 F9 2A 19 F9 D9 : 6D
--------------------------------
SUM: 0C 85 A8 34 45 41 30 4E 8B4B

F900 C3 1B F9 D9 ED 5B 17 F9 : 08
F908 2A 19 F9 D9 C3 A6 F9 D9 : 50
F910 2A 19 F9 D9 C3 29 FA 00 : FB
F918 00 00 00 F5 C5 D5 E5 3A : AE
F920 A4 FA 4F D9 08 7D 08 D9 : 2C
F928 3A A3 FA 47 D9 08 6F 08 : 76
F930 D9 79 FE 17 D2 94 F9 B7 : 7D
F938 CA 94 F9 CD 93 FA CB F4 : 70
F940 78 FE 1F D2 82 F9 B7 CA : 63
F948 82 F9 16 08 E5 D9 C1 D9 : F1
F950 D9 1A ED 79 13 1A CB F8 : 49
F958 CB B0 ED 79 13 1A CB F0 : C9
F960 ED 79 CB B8 13 3E 08 80 : C2
F968 47 D9 15 C2 50 F9 04 23 : 67
F970 D9 2D D9 C2 40 F9 0C D9 : BF
F978 25 D9 C2 28 F9 E1 D1 C1 : 54
--------------------------------
SUM: 68 10 B5 B4 A7 29 21 60 38FE

F980 F1 C9 04 CD 93 FA CB F4 : D7
F988 D9 E5 21 18 00 EB 19 EB : E6
F990 E1 C3 71 F9 0C D9 E5 7D : 55
F998 21 18 00 EB 19 3D C2 9C : D8
F9A0 F9 EB E1 C3 78 F9 F5 C5 : B3
F9A8 D5 E5 3A A4 FA 4F D9 08 : C2
F9B0 7D 08 D9 3A A3 FA 47 D9 : 55
F9B8 08 6F 08 D9 79 FE 19 D2 : BA
F9C0 17 FA CD 93 FA CB F4 78 : A2
F9C8 FE 28 D2 05 FA 16 08 E5 : FA
F9D0 D9 C1 D9 D9 ED 78 12 13 : D6
F9D8 CB F8 CB B0 ED 78 12 13 : C8
F9E0 CB F0 ED 78 12 CB B8 13 : C8
F9E8 3E 08 80 47 D9 15 C2 D3 : 90
F9F0 F9 04 23 D9 2D D9 C2 C7 : 88
F9F8 F9 0C D9 25 D9 C2 B3 F9 : 4A
--------------------------------
SUM: D3 B3 3E 21 05 87 C8 99 AA42

FA00 E1 D1 C1 F1 C9 04 CD 93 : 91
FA08 FA CB F4 D9 E5 21 18 00 : B0
FA10 EB 19 EB E1 C3 F4 F9 0C : 8C
FA18 D9 E5 21 18 00 EB 7D 19 : 78
FA20 3D C2 1F FA EB E1 C3 FB : A2
FA28 F9 F5 C5 D5 E5 F3 01 03 : 64
FA30 1A 3E 0B ED 79 3D ED 79 : 6C
FA38 3A A2 FA 4F D9 08 7D 08 : 8B
FA40 D9 3A A1 FA 47 D9 08 6F : 45
FA48 08 D9 79 FE 19 D2 8E FA : CB
FA50 CD 93 FA 78 FE 28 D2 86 : 50
FA58 FA 16 08 E5 D9 C1 D9 D9 : 49
FA60 AF ED 79 3E 08 80 47 D9 : FB
FA68 15 C2 5F FA 04 23 D9 2D : 5D
FA70 D9 C2 53 FA 0C D9 25 D9 : CB
FA78 C2 41 FA FB 01 00 40 ED : 26
--------------------------------
SUM: 30 9F EB 50 E3 2D 4F CB 1FAA

FA80 78 E1 D1 C1 F1 C9 04 CD : 76
FA88 93 FA D9 C3 6F FA 0C D9 : 77
FA90 C3 76 FA 26 00 69 54 5D : 73
FA98 29 29 19 29 29 29 58 19 : 57
FAA0 C9 00 00 00 00 00 00 60 : 29
FAA8 00 C0 00 20 01 80 01 20 : 82
FAB0 01 E0 01 40 02 A0 02 40 : 06
FAB8 02 F5 C5 D5 ED 5B D2 FA : A5
FAC0 01 83 03 CD D5 FA 22 D2 : 17
FAC8 FA ED 5F 32 D4 FA D1 C1 : D8
FAD0 F1 C9 33 E9 00 21 00 00 : F7
FAD8 79 B0 C8 CB 38 CB 19 30 : 08
FAE0 01 19 CB 23 CB 12 18 F0 : ED
FAE8 3E FF 32 C0 F8 C3 73 ED : 4A
FAF0 C5 E5 6F 0E 0A CD 0C FB : 05
FAF8 7D B7 28 02 C6 10 C6 20 : 1A
--------------------------------
SUM: A9 AC 74 AE ED 62 FA 91 39CF

FB00 CD 13 00 7C C6 30 CD 13 : 32
FB08 00 E1 C1 C9 26 00 06 08 : 9F
FB10 29 7C 91 38 02 2C 67 10 : 13
FB18 F7 C9 21 A6 FC CD 4B FB : 96
FB20 AF 32 A5 FC CD 6E FB C3 : 7B
FB28 8B FB 3E 01 32 A5 FC 21 : B9
FB30 B5 FD CD 4B FB 18 ED 3E : 08
FB38 02 32 A5 FC 21 CE FD CD : 8E
FB40 4B FB 18 E0 16 07 3E FF : 98
FB48 C3 8F FC 22 A1 FC 22 A3 : D2
FB50 FC 21 00 00 22 9F FC 22 : FC
FB58 9D FC 3E 02 32 9C FC 32 : D5
FB60 9A FC 3E 01 32 9B FC 3E : DC
FB68 FC 16 07 C3 8F FC 3A A5 : 46
FB70 FC B7 FA 1E FC 3A 9F FC : 9C
FB78 47 3A 9D FC B8 CC A8 FB : 41
```

```
--------------------------------
SUM: 5E 3F F6 49 85 FD 3B E5 9B54

FB80 3A 9B FC 3D CC 3B FC 32 : 43
FB88 9B FC C9 3A A5 FC B7 FA : EC
FB90 1E FC 3A A0 FC 47 3A 9E : 0F
FB98 FC B8 CC CD FB 3A 9C FC : 1A
FBA0 3D CC 4B FC 32 9C FC C9 : E3
FBA8 2A A1 FC 7E 23 4E 23 46 : 1F
FBB0 23 16 00 CD 8F FC 14 79 : 1E
FBB8 CD 8F FC 78 32 9F FC 22 : BF
FBC0 A1 FC 3E 0F 16 08 32 9D : D7
FBC8 FC CD 8F FC C9 2A A3 FC : E6
FBD0 7E 23 4E 23 46 23 16 02 : 93
FBD8 CD 8F FC 14 79 CD 8F FC : 3D
FBE0 78 32 A0 FC CD F5 FB 22 : 25
FBE8 A3 FC 3E 0F 16 09 32 9E : DB
FBF0 FC CD 8F FC C9 3A A5 FC : F8
FBF8 FE 01 28 17 FE 02 CA 30 : 38
--------------------------------
SUM: 43 D4 BA 03 C6 99 CE F3 F2F1

FC00 FC E5 11 B4 FD B7 ED 52 : 99
FC08 E1 30 01 C9 21 A6 FC 22 : C0
FC10 A1 FC C9 E5 11 CD FD B7 : DD
FC18 ED 52 E1 30 01 C9 3E FF : 57
FC20 16 07 CD 8F FC 32 A5 FC : 48
FC28 C9 3E FE 16 07 C3 8F FC : 70
FC30 E5 11 EF FD B7 ED 52 E1 : B9
FC38 30 E4 C9 3A 9D FC 3D 32 : 1F
FC40 9D FC 16 08 CD 8F FC 3A : 49
FC48 9A FC C9 3A 9E FC 3D 32 : A2
FC50 9E FC 16 09 CD 8F FC 3A : 4B
FC58 9A FC C9 3E A6 16 04 CD : 2A
FC60 8F FC 3E 06 C3 6F FC 3E : 3B
FC68 64 16 04 CD 8F FC AF 14 : 99
FC70 CD 8F FC 3E F8 16 07 CD : 78
FC78 8F FC 3E 10 16 0A CD 8F : 55
--------------------------------
SUM: 1D 2A 79 18 C5 8C 9F 56 2479

FC80 FC 3E 10 14 CD 8F FC 3E : F4
FC88 10 14 CD 8F FC AF 14 C5 : 04
FC90 01 00 1C ED 51 05 ED 79 : C6
FC98 C1 C9 00 00 00 00 00 00 : 8A
FCA0 00 00 00 00 00 00 FD 00 : FD
FCA8 0B 51 01 0B A9 01 0E 7B : 9B
FCB0 01 0B 7B 01 0C A9 01 0E : 4C
FCB8 7B 01 0D 65 01 0A FD 00 : F6
FCC0 0B 51 01 0B A9 01 0E 7B : 9B
FCC8 01 0D 7B 01 0C A9 01 0E : 4E
FCD0 7B 01 0D 65 01 0A 51 01 : 4B
FCD8 0E FD 00 0B FD 00 0B E1 : FF
FCE0 00 0B E1 00 0B D4 00 0B : D6
FCE8 D4 00 0C FD 00 0B FD 00 : E5
FCF0 0E E1 00 0C D4 00 0E E1 : BE
FCF8 00 0B FD 00 0B 1C 01 0B : 3B
--------------------------------
SUM: CC CB F5 86 6D A6 7D 67 2520

FD00 3E 01 0B 51 01 0B 3E 01 : E6
FD08 0C 1C 01 09 1C 01 0E FD : 5A
FD10 00 0B FD 00 0B E1 00 0B : FF
FD18 E1 00 0B 1C 01 0B 1C 01 : 31
FD20 0C FD 00 0B FD 00 0B 51 : 6D
FD28 01 0E 7B 01 0B 3E 01 0B : E0
FD30 FD 00 0B D4 00 0B E1 00 : C8
FD38 07 FD 00 07 E1 00 00 FD : E9
FD40 00 08 51 01 0A A9 01 0E : 1C
FD48 7B 01 0B 7B 01 0C A9 01 : B9
FD50 0E 7B 01 0C 51 01 07 FD : EC
FD58 00 0B 51 01 0A A9 01 0E : 1F
FD60 7B 01 0C A9 01 0E 7B 01 : BC
FD68 0C 51 01 08 51 01 0E FD : C3
FD70 00 0B FD 00 0B E1 00 0B : FF
FD78 E1 00 0B 51 01 0B 51 01 : 9B
--------------------------------
SUM: 2D 1C 5D E8 D6 9B E1 87 07A8

FD80 0C FD 00 0A 0C 01 0E FD : 2B
FD88 00 0C E1 00 0E D4 00 07 : D6
FD90 BD 00 0C D4 00 0D E1 00 : 8B
FD98 0C FD 00 0E E1 00 0B D4 : D7
FDA0 00 0C E1 00 0E FD 00 0C : 04
FDA8 0C 01 0E 2D 01 0C 1C 01 : 72
FDB0 0E FD 00 00 00 47 00 0D : 5F
FDB8 3F 00 0D 47 00 0D 3F 00 : DF
FDC0 0D 47 00 0D 3F 00 0D 3B : E8
FDC8 00 0A 00 00 0A 00 DD 01 : F2
FDD0 0D 7D 02 0D DD.01 0D 7D : 01
FDD8 02 0D DD 01 0D A3 02 0D : AC
FDE0 BB 03 0A 23 03 0D 85 03 : 83
FDE8 0D BB 03 0B 00 00 0A 00 : E0
FDF0 03 77 53 5A 56 77 CE 9F : 61
FDF8 EB 2E CE 9F D3 15 FF FF : 6C
--------------------------------
SUM: 00 4E F6 A2 69 7C AA 59 19FF
```

Q.A Oh!MZ 質問箱

Q **MZ-2500を使用しています。POKE@ を用いてアスキーコードから対応するキャラクタを表示させ，PEEK@を用いてキャラクタから対応するアスキーコードを得る方法を教えてください。たとえば，WIDTH40でLOCATE10，10：PRINT“！”と同じ動作をさせようとしてPOKE@&H38，410，33とするとBSと表示され，「！」を表示するには，POKE@&H38，410，33＊4としなければなりません。一方，LOCATE10，10：PRINT“A”としてA＝PEEK@（&H38，410）とすると，A＝65ではなくA＝4となります。　東京都　本宮　卓**

直接キャラクタVRAMをアクセスしてPRINTやSCRN$に相当することをしてみたいというわけですね。

MZ-2500ユーザー以外の方のために，簡単に状況を説明しますと，POKE@とPEEK@は任意のメモリブロック（全メモリを8Kバイトごとに区切ったもの）のオフセットアドレスに値を書き込んだり読み出したりする命令です。質問にあるPOKE@文はメモリブロック38_Hの410番目のアドレスに33を書き込むという意味になります。メモリブロック38_Hはテキストおよびアトリビュート用RAMになっており，40×10＋10番目のメモリに“！”のアスキーコードである33を書き込んで座標（10，10）に“！”を表示しようというのが本宮さんの狙いなのです。

ところがこの試みはうまくいきませんでした。MZ-2500のテキストVRAMに書き込まれているのはアスキーコードではなくフォントが格納されている「漢字ROM内のオフセットアドレス」だからです。半角文字のフォントも漢字ROMに入っているのです。

オーナーズマニュアルの「MB38，MB39の詳細」と題されたページを見てください。このページには致命的な記述落ちがあるものの，大まかなヒントを得るには十分な説明がなされています。まず，テキストVRAMは800_Hバイトごとに4つのブロックに分けられ，それらは順にテキスト1，アトリビュート，テキスト2，未使用といった使われ方をしているのがわかります。未使用部分を除く3つのブロックすべてにそれぞれ値を書き込むことにより1文字が表示されるのです。（0，0）の位置に表示したければオフセットアドレス0000_H，0800_H，1000_Hに値を入れることになりますね。アトリビュートはマニュアルのとおりですから重ねて説明はしません。単に白で表示したければ7を入れておくだけですし，普通の状態では（画面をクリアしても）そうなっていますからいじる必要さえないでしょう。問題はテキスト1と2に書き込む値です。

視点を変えて漢字ROMの中がどのようになっているのか考えてみましょう。MZ-2500には標準でJIS第1水準，第2水準それぞれ128Kバイト，計256Kバイトの漢字ROMが搭載されています。16×16ドットの文字フォントは32バイトになりますから，32バイトを1文字分として格納されていると想像されます。すでに述べたように実際には半角のフォントも入っているわけですから，8バイト（8×8ドット分）を最小単位として，2つの連続した領域で8×16ドットの文字を表し，4つの領域で16×16ドットの文字を表すようになっているはずです。最小単位が8バイトですから，任意の文字フォントが格納されているアドレスを指定するには18ビット（256K）は必要ではありません。最後の3ビットは常に0になるからです。よって，上位の15ビットでアドレスを特定することができます。

このような予想を立てたうえでVRAMを振り返ってみますと，テキスト1と2を合わせれば16ビットになりますから，そのうち15ビットで漢字ROMアドレスを指定し，残った1ビットでPCGか漢字ROMかを選択することができます。しかしながら，マニュアルの記述はそうなっていませんでした。テキスト1の最下位ビットはdon't care（0でも1でもよい）で，第1ビットは0か1かで16×16ドットの右側か左側かを表し，残り6ビットで漢字ROMアドレスを指定すると書いてあります。また，テキスト2は第5ビット以下が空欄で，第6ビットはアクセスする漢字ROMが第1水準か第2水準かを表し，最上位ビットでPCGか漢字ROMかを指定するという表現になっています。一瞬とまどいましたが，テキスト2の空欄に「漢字ROMアドレス上位」が入ることに気がついてようやく納得がいきました。

マニュアルの図をクリーンアップしたのが図1です。予想したとおりテキスト1が漢字ROMアドレスの第3～10ビットを，テキスト2の最上位ビットを除く7ビットがROMアドレスの第11～18ビットを表します。テキスト1の第1ビットで漢字の左か右を表すというのは，このビットが0か1かで漢字ROMアドレスが16バイトずれるという意味（マニュアルの記述は逆）に解釈できますし，テキスト2の第6ビット

図1　VRAMの構成

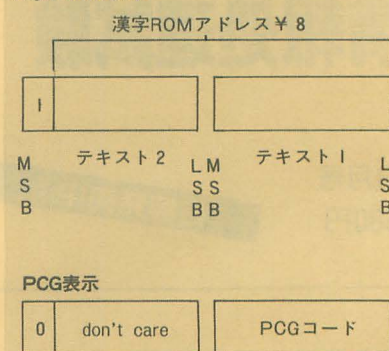



表1　don't careの条件

画面桁数 ＼ フォント	12	20	25
8×8	○	○	○
8×16	○	×	×

で第1水準と第2水準を切り換えるというのは漢字ROMが128Kバイトごとの2つに分けられていることと一致します。

マニュアルでdon't careとなっているテキスト1の最下位ビットも「必要な場合には漢字ROMのアドレスを8バイト単位で指定するという働きをします。どんな画面モードのときにdon't careであるのかを表1にまとめておきます。×がdon't careで，その場合は1にしても0とみなされます。○はROMアドレスの第3ビットとして意味を持つモードです。

この表から面白い事実が見えてきます。たとえば8×8フォント25行のモードでは通常漢字の上半分しか表示されませんが，直接VRAMにROMアドレスを書き込んでやれば下半分も表示できることがわかります。さらに，12行のモードではちょうどこれと同じような処理をしているだろうことも予想できます。これはハード的には12行のモードは存在せず，ソフト側で25行モードの2行をまとめて1行のように扱っているということを意味します。

さて，やっと文字表示の理屈がわかったところで問題の漢字VRAMのアドレスを求めることにしましょう。リスト1が実験用に作ったプログラムです。このプログラムは漢字ROMの先頭から順に倍角512文字ずつ（32×512＝4Kバイト）表示するものです。同時に左上隅に表示している文字の漢字ROM内アドレス(水色で表示)とテキスト1および2に入れる値を2進数で表示します。普段は使えない（文字コードが与えられていない）文字も表示されますので，MZ-700用の文字フォントや，16×8ドットのひらがな・各国のアルファベット・発音記号，16×16ドットの縦書き用カッコなどという見慣れないものを見ることもできます。

このようにして漢字ROMの中身を覗いてみますと，目指す半角文字のフォントは漢字ROMの6000_H番地から$7FFF_H$番地に，8×16ドットフォント（16バイト），2500モード用8×8ドットフォント，2000モード用8×8ドットフォント（それぞれ8バイト）の計32バイトが1組となりアスキーコード順に並んでいます。よって任意のアスキーコードに対応する漢字ROM内アドレスは，8×16ドットフォントなら6000_H＋アスキーコード×32，2500モード用8×8ドットフォントであれば6000_H＋アスキーコード×32＋16，2000モード用8×8ドットフォントなら6000_H＋アスキーコード×32＋24という式で求めることができます。

あとは図1に示したようにVRAMに書き込めばその文字が画面表示されるわけです。具体的には上の計算式によって求めた漢字ROMアドレスを8で整数除算を行い，さらに256で割った余りをテキスト1へ，商に80_Hを足した値(テキスト2の第7ビットを1にする）をテキスト2に書き込めばよいのです。“A”を表示したければテキスト1に04_Hを，テキスト2に$8D_H$を書き込みます。画面を読んでアスキーコードを得たければちょうど逆の変換をすることになりますね。

ついでですから全角文字の表示についても触れておきましょう。上記の手順で漢字の左半分を表示することができます。残り右半分はVRAM上のすぐ次のアドレスに，テキスト1については左半分の値に2を加え，テキスト2は左半分と同じ値を書き込むようにします。また，これだけですと，表1で○が付いているモードでは全角文字の上半分しか表示されませんから，下側も表示したければ次の行に対応するアドレス（画面の横文字数を足したアドレス）にテキスト1については上の行で書き込んだ値に1を加え，テキスト2はそのままの値を書き込むという作業を左側右側の両方に対して行います。

ここからはまったくの余談になりますが，2000モードで動作しているときにVRAMに書き込まれるのはアスキーコードです。漢字ROMのアドレスを入れる構造のVRAMにアスキーコードを入れても文字が表示されるはずがないと思って少し調べてみたのですが，どうやらあれはPCGのようです。IPLが漢字ROMからMZ-2000の文字フォントを読み込んでPCGとして定義し，アトリビュートとテキスト2の設定を行ったうえで2000モードに切り換えているのです。あとは2000のVRAMが置かれているアドレスにメモリブロック38_Hが割り当てられると思えば2500モードでPCGを表示するのと変わらないというわけです。

（瀧山　孝）

リスト1　テストプログラム

```
10 init "crt:80,25,1,0"
20 locate 0,15:print "-TEXT 2- -TEXT 1-"
30 print "1";:color 5:print "0000000 00000000 000"
40 for I=0 to 1023
50   poke@ &H38,I,0
60   poke@ &H38,I+&H800,7
70   poke@ &H38,I+&H1000,&H80
80 next
90 for I=0 to 127
100   for N=0 to 7
110     poke@ &H38,N*128+I,I*2
120   next
130 next
140 'for J=0 to 63 step 8          第２水準漢字rom無
150 for J=0 to 127 step 8        : 第２水準漢字rom有
160 locate 1,16:print right$("000000"+bin$(J),7)
170   for I=0 to 127
180     for N=0 to 7
190       poke@ &H38,N*128+I+&H1000,N+J+&H80
200     next
210   next
220   A$=input$(1)
230 next
240 color 7
```

質問にお答えします

日ごろ疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集室が総力をあげてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに回答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に機種名，システム構成，必要なら図も入れてこと細かに書いてください。また，返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について，直接問い合わせることもありますので，電話番号も明記してくださいね。

宛先：〒102 東京都千代田区
九段南2-3-26井関ビル
(株)日本ソフトバンク出版部
「Oh！MZ質問箱」係

STUDIO MZ

FROM READERS TO THE EDITOR

朝の4時というともう明るくなる季節です。夏バテせずにお過ごしですか。日焼けで顔が黒光りしてる人もいるんでしょうね。今月は，すっかり「その筋」している読者の皆さんによる，5周年記念と「その筋事典」へのお便りを紹介しましょう。

◆QDはシャープのサービスセンターで部品として取り扱っています。これはシャープのOA相談室からの返答です。　笹井　浩行（18）福井県

シャープはQDユーザーを見捨ててなかった！　わけだけど，も少し積極的にやってくれませんかねぇ。

◆中野修一さん。私もまったくそのとおりだと思うのです。いま，私がいちばん必要としているのは強力なBASICコンパイラなんです。求む，究極のBASICコンパイラ！　浜本　悟（32）鳥取県

6月号にはストロングタイプの読者によるFuzzyBASICコンパイラ登場。メーカーを引っ張るのはユーザーの皆さんです。

◆じつはあさってから中間試験なんです。そこで僕は「気合い」を入れて勉強しよう！　と思いながら愛読者カードを書いています。そして，「これを書き終えたら数学だ」と思いながらパソコンの電源を入れてしまう今日このごろです。もうだめだ——。　山下　祥宣（16）千葉県

「BASICで数学と遊ぶ」プログラムもあります。電源入れたらさァ勉強。

◆うれしー，「アルバトロス」でホールインワンをきめたんだい。CONGRATULATIONSの表示が出て音楽がピロピロ鳴ってたんだよねー。そのあとも調子に乗って，その日は18ホールで－10という自己最高記録を樹立したのだよ。いまのところ，友だちの中では誰にも破られていないぜ。

楠本　靖幸（20）愛知県

10アンダーぐらいはまだまだヒヨッコ。世の中にはもっとすごいツワモノがいるはずです。

◆仕事上の勉強のためということでX1turboZをヨメさんに買ってもらいました。さあて，次はなんといってX68000を買ってもらおうかな——。それにしてもX1turboZはすごい。この値段ではこれだけの機能があるとは……。感激です。

日並　輝秀（29）福岡県

ディスプレイはX68000と共用できるし。これは奥さまもお買い得と思うでしょう。

◆50の手習いで息子のパソコンに手を出し，ポチポチと半英語のBASICを入力して遊んでいます。MZ-2500用のマシン語を解説してある本があったら教えてください，よろしく。　林　章（49）

入門からちょっとした応用まで解説書は数冊出てます。本屋さんで探してみてください。

◆つい先日，ディスクのFATとディレクトリテーブルを壊してしまった。マシン語をさわっていたら暴走しちゃったんです。そこで，DREAMという素晴しいディスクエディタがあることを思い出し，先頭クラスタやクラスタのつながりに苦しみながら，なんとか蘇生させることができました。すごいなあ，S-OSバンザーイ。

竹山　志朗（17）石川県

皆さんが使いこなしてくれてこそS-OSも生きてきます。

◆シャープはもっと前からMZ-2500用の増設RAMなどを値下げするべきだった。MZ-2520なんか出さなければ，増設RAM，VRAM対応ソフトがたくさん出たことだろうに……。かくいう私はコンパチのRAMボードをVRAMも含めて14,800円で買ったのである。　金子　智裕（17）神奈川県

手に入れやすくなった増設ボード。後手に回ることも多いメーカーサポートだけど，努力のあとは認めてあげましょう。

◆昔，Oh！MZはアダルトブックのコーナーにあったこともあった（オークスターのお姉さんが表紙を飾ってたころ）が，いまだにOh！FMを音楽関係のコーナーに置いてる本屋もある。

北野　明（22）大阪府

たぶんFMサウンド特集の号だったんですよ。CG特集したらアニメのコーナーへ移ったりして……，本屋さんもお茶目ですね。

◆韓国で中高校生の憧れとなっている三星電子製8ビットパソコン「SPC-1500」というマシンは，日経パソコン6月8日号217ページの記事によると，なんとX1クローンなのだそうだ。外見はX1Cにそっくりだが，カセットデッキ部に押しボタンが5つも並んでいるところを見るとフルロジックコントロールではないのかな？　いずれにしても外国で互換機が作られるとは，X1もIBM並みにメジャーになった？　城　昌克（32）愛知県

ハングル語版のZ'sSTAFFや即戦力なんかも出るんでしょうか。面白そうだなあ。

◆「俺は聖帝サウザーだ，レーザーもオプションもいらぬわ！」と言ってX68000やアーケード版のグラディウスを一周してしまう僕は，ひょっとしてどこかおかしいのでしょうか？

野宮　行治（20）岡山県

ご心配にはおよびません。誰にだって隠し技のひとつやふたつあるもんです。

◆愛しのX1CkにFDDのCZ-503Fがついてはや1カ月。明日から中間考査だというのにウィザードリィに明け暮れてます。ウイバーンもやってたけど，マップが広すぎて2面でめげてしまいました。それからね，X1のグラディウスで裏ワザを使うと，な，なんとオプションが4つ付くのだ。さすがはX1だぜ。それにしてもディスクはうるさい。　山本　伸明（15）北海道

オプション4つにすると強くなりすぎて面白くないって人もいました。あはは。

◆あのー，数学で{Y〔Z+(X+Y)〕}^{2}という式をスーパーかっこを使って（Y（Z+(X+Y〕^{2}のように書いてしまったり，ノートをまとめるときに字下げをして構造化をしようとする僕はいったいなんなのでしょうか。　上村　圭介（15）埼玉県

Oh！MZの読者でしょ，もちろん？

◆私は創刊号を持っていません（読んだことはありますが）。じつをいうとあの表紙に驚いてしまったのです。そしてX1の記事が出るところまでは立ち読みの日々が続きました。しかしある日，バックナンバーを揃えようと思い，創刊号から8冊をある本屋で注文したのですが，4年たったいまでもその本屋からはなんの連絡もありません。本屋のバカヤロー。これはノンフィクションです。

西塚　公貴（18）埼玉県

本屋さんものん気すぎるけど，催促はしてみなかったんですか？

◆あれは5年前の12月だった。私はN社のパソコンを買うべく店に行った。だがそこで私が見たものは，カラフルなパソコンテレビであった。それ以来，私はこいつと付き合っている。こいつ，シャープのCZ。たいしたやつだぜ！

古村　康英（18）長野県

やっぱりいいものは長く使えますね！

◆飼い猫がプログラミングのときにじゃまをする，といってる人がいますが，うちのルル（ペルシャのメス）は違います。MZ-1500のQDを起動するときにちゃんとEJECTボタンを押してくれるんです。といってもたまにですが。じゃましたことは一度もありません。せっせと僕がキーボードをたたいていると肩の上に乗るので，寒いときなど暖かくてとても気持ちいいんですよ。やっぱりペットは猫がいちばんです。山本　康博（18）長崎県

ユーザースペシフィックなオートイジェクトキャットという周辺ペット。なんて可愛いの。

◆いま，夢に凝っている。眠っているときに見るアレである。友人と話をしていて気づいたのだが，夢にもグレードがあるようだ。つまり，カラーだったり音が聞こえたり，という差である。僕の夢

藤井　雅子（18）福岡県

はどんなものかとゆーと，自分の目で見，触るとしっかり手応えがあるのだ。この機能のおかげでその筋の夢を見た日は気分がよいのである。いいでしょう。友人たちは僕を変態といってうらやましがっている。そこで質問。あなたの見る夢はどんな夢ですか？　　木内　英明（17）愛知県

触覚つきの夢ですか。あとは味覚もあれば完璧ですね。私はたまに，前日見た夢の続編を見たりします。

◆祝氏に，西伊豆のカタスミにもファンがいることを，もしこのハガキが目に留まったら伝えていただきたいと思います。ファンというには少し年をとりすぎてるんですが。

橋本　幸一（38）静岡県

しっかり伝えます。カタスミといわず西伊豆代表のファンというのはいかがですか？

◆「うる星やつら」も終わった。「めぞん一刻」も終わった。これからはOh！MZだけが青春です。

安岡　賢太郎（16）神奈川県

あなたの青春は明るい！

◆近所のパソコンショップにX68000とPC88VAが同時に並べられた。PC88VAはちゃんとしたデモをやっていたが，X68000は隣でグラディウスだけをやっていた。そんなところを見たらPC88VAのほうが性能もよく見える。そのうえ，あのCMを見たら真実を知らない人間はPC88VAを買うに違いない。しかも店員はNECの製品ばかりを客にすすめる。ということは，X68000は最初からそれを目当てに来た人しか買わないのだろうか。

座喜味　聡（17）沖縄県

失敗を繰り返して人間は成長します。X68000のよさはいずれわかるでしょう。

◆6月号の表紙の仮面を見て「URYYY！」などといってしまったのは私だけだろうか。これは，「コンパイラで未知なる力を引き出せ」という意味なのか。私はいたずらに血を吸わせてゾンビハッカーを作るだけだと思ったりするが……。清水さん，やってくれましたね。私はかねがねこういう昔のソフトを待っていたのです。でも毎月やるとネタが尽きてしまうだろうから(尽きなかったりして)隔月くらいで載せてください。最近のゲームにはない新鮮さがあるから。後藤　裕治（21）大分県

マシンの可能性と活用を考えるのがOh！MZの目的です。ゾンビじゃありません，きわめて健康にいいですよ。

◆マシン語を始めて4年近くになりますが，「マシン語開発プログラム入門」はたいへん面白かった。いまだにフロッピーもプリンタもない私は，登内敏夫氏の記事にあった，むかし話の丸貧じいさんに限りない愛着を持ってしまった。やはり，マシン語開発にはハングリー精神が大切である。

横田　紀明（20）山口県

逆境に強いのがパソコンユーザーの特徴です。

◆X68000をクレジットで買いました。支払いは8月10日の一括払いです。そこでX68000用に貯めたお金で株に手を出したら8万円もうかってしまいました。X68000は40万円まで下がったことになる？　　森井　巌（28）愛知県

Oh！MZの読者にも資金運用の上手な人いるんですね。8万円でとどまったのは偉いなあ。

◆その筋キーホルダーの申し込みは「官製ハガキで」となっていますが，愛機X1Cをかたわらに毎月3千円近く払ってここケニアで購読している私には応募ができないではないですか。絵ハガキでもいいことにしてください。

Naoto Noda（29）Nairobi

遠路はるばるのお便りありがとうございます。その筋キーホルダー送らせていただきます。ところで，ケニアでもX1Cは動くんでしょうか？

◆大学院に入ったらヒマになるはずであったが，なぜか相も変わらず忙しい毎日である。困ったことだ。その忙しさの中で，講義でLISPをやり，研究でPascalを使用し，家では趣味でCを，といままで使ってきたBASIC，Z80Assembler，FORTRANに加えて3つもいっぺんに習得しようと考えている今日このごろである。　戸泉　厚（23）京都府

マルチリンガルとはあなたのこと。Oh！MZのスタッフには母国語はマシン語だという人もいまして。

◆ほんと！　6月号に載っていた読者の声にあるように，MZ-2500のようなマシンを世間の人は知らないんでしょうかねぇ。私などはとことん入れ込んでいます。ゲームもほとんど揃えてるし……。といってもゲームはあまりやらず，BASIC，CP/M，S-OSを起動しているほうが多いなあ。おっと，テレホンソフトも忘れてはいけない。MZ-2500ってなんて守備範囲が広いんだろう。世間のほうがよっぽど視野が狭いですね。田中　充（23）大阪府

コストパフォーマンスはユーザーが高めるものでもあるんですね。これからもその守備範囲を広げていってください。

◆X68000の宣伝について提案があります。菊池桃子と明石家さんまを日立に，中森明菜をパイオニアに，西村知美を東芝に，そして斉藤由貴を日電にとられた現在，シャープを支えるだけの力を持っている関西のタレントは，そうです。南野陽子ちゃんです！　彼女と西川のりお，これぞ異色のコンビ。これに関西大学の林省之助教授が加われば，日電は確実に倒せる！　もちろん荻野目洋子ちゃん，金田一先生にはMZ書院のほうでがんばってもらう。これで決まりだ。

福島　義浩（18）滋賀県

異色のコンビでCMも出色のできばえ，とうまくいきませんかね。

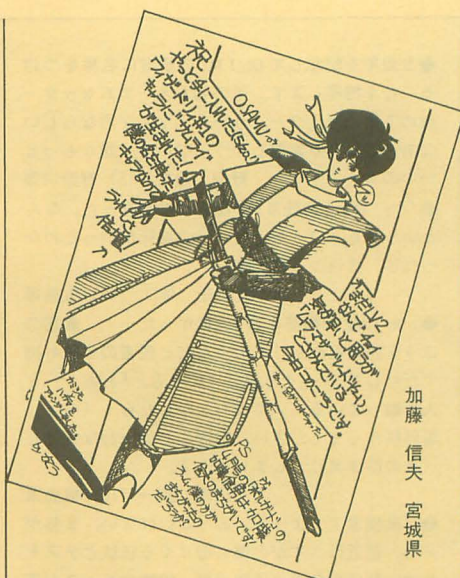
◆先日，初めてX68000にご対面できました。しかもNECのショップでです。ビジュアルシェルが見たかったのに，X68000にへばりついているおじさんは，グラディウスに熱中していて当分動きそうになかったため，あきらめて帰りました。どうしてNECのファンはゲームが好きなんでしょうか。

松野　裕之（20）徳島県

まあ抑えて。そのグラディウスおじさんがX68000ユーザーになれば仲間が増えるし。

◆僕は発見した。なんとあの祝氏の島をである。山口県徳山市の南，瀬戸内海にぽっかりと「祝島」の文字が！　ちなみに隣は「長島」だ。うーん，こんな島を見つけてしまってはSTUDIO MZに載るのは目に見えてるな。やぁみんな，読んでるかぁ，オレだぞぉ。　　馬場　啓示（14）宮城県

その昔，瀬戸内海を荒らした海賊を当時の武将千平(せん)が退治してそのほうびに島を賜った，という史実は……なさそうですね。

◆MACINTO-Cをお作りになった泉大介先生，お願いがあります。どうか，縦横サムとCRCだけ入力すれば128Kバイト入力したことになるような，錬金術プログラムを作ってください。僕を助けると思って。　　渡辺　智樹（16）栃木県

加藤　信夫　宮城県

さすがのスーパーマン泉氏も錬金術プログラムにはうなってしまうのでは……。

◆昨年の終わりごろ，Oh！MZに，サムシンググッドからワープロソフトShogunが「近日発売予定」だという記事が載ったので楽しみに待つこと半年。サムシンググッドに電話を2度もかけてしまった。待ちきれずにSamuraiを買った。いつまでたっても「近日発売予定」では疲れますよ。

浅井　勝（42）愛知県

やっと「近日」の頭が見えてきたようです。待ちくたびれちゃいましたね。

◆6月8日は中学生になって初めての誕生日。なのに父は我が愛機MZ-1500に封印をすると言ってきかない。なんとかして……！

花岡　宏之（12）大阪府

封印されるような要因をまず取り除かねばなりますまい。

◆暑い日々がやってきたようですね。我が家のX1につながっている専用ディスプレイにも夏がやってきたようです。スイッチを入れたあと，1時間ほどたつと「プチン」という音とともに画面が勝手に消えてしまい，それからは何度スイッチを押してもなんの応答もないのです。注意深くスイッチ付近を見ますと，なんとランプは赤でオフになっているはずなのに，電気のメーターはくるくる回っているんです。このポーカーフェイステレビ，やはりシャープに問い合わせるべきでしょうか。

松本　吉紀（18）東京都

やたらに暑かったある夜，ポータブルCDが暴走して私の手の中で踊り出しました。暑気当たりが軽いうちに早いとこ修理を。

◆新しいマシンが次から次と目まぐるしく，それを追いかける若者がうらやましい。いったいどんなことに活用しているのだろう。MZ-2000も完全に使いこなせていない小生などは，この愛機と生涯を共にすることであろう。かめばかむほど味の出るスルメのような存在である。

下山　芳次郎（62）大阪府

愛機と生涯を共にできるなんてうらやましい。かむほどにどんな味がするんでしょうね。

◆1年前Oh！MZを読んでもチンプンカンプンでした。最近は半分理解できるようになり，毎月18

◆5周年を記念してOh！MZの読者に名称をつけることを提案します。その名もオーエムゼッター。縮めてオエゼッター。うーん，なんか汚ならしいなあ。これじゃ定着しそうにないや。誰かもっといいの考えて！　野田　敏之（15）神奈川県

◆「Oh！MZその筋事典」は非常に役立った。なんといっても「ローディスト」の意味がやっとわかったし。面白かったです。
多田　健二（16）広島県

◆「Oh！MZその筋事典」面白かったー！　最初のほうを読みながら，「このぶんだと読者の名前も出てたりして……」そう思っていると，「おお！　2人も載っている！」ではもしかしたら！「……」。番外編作ってください。無理にとはいわないけど。「その筋事典II」しましょうよ。
山崎　潤一（18）福島県

◆「高橋君ですよ――」おっといけない，まちがえた。最近はさすがの僕も迫りくる模試とテストと赤い数字の脅威には勝てず，投稿のペースが落ちてしまいました。ま，いいさ，少数精鋭でいこ（うわぁ，キタナいイラスト！）。ところで5月号では「Oh！MZその筋事典」に載せていただき，どうもありがとうございました。しかし，あれでは，“僕はローディストですよぉ”って公言してしまったようなもんですね。中森さんがこんなに“その筋”だなんて知らなかった。ファンタニカに載るな，こりゃ。ところで，確かに「ファンロードでは毎回没」なんですけど，いっぺんだけ載ったことあるんですよ。　高橋　哲史（17）福岡県

◆「Oh！MZその筋事典」がメチャ面白かった。笑いながらOh！MZの歴史を学べるとは，すごい企画だ！　渡部　秀剛（18）滋賀県

◆創刊5周年おめでとうございます！　これから10年，20年，100年とがんばってください。
池田　雄一郎（14）福島県

◆創刊5周年おめでとうございます。創刊号というのは薄くてもいまより高く620円もしたんですね。驚き驚き！　これからも内容は充実，値段は安くの精神でお願いします。
上林　清（32）千葉県

◆5周年おめでとうございます。息子も5月18日に5歳の誕生日を迎え，元気に幼稚園に通っています。15年後，揃って成人式を祝いたいものですね。　山本　雅昭（31）神奈川県

◆ギャー！　Oh！MZはなんてことをしてくれたんだ！　試験前だというのに創刊5周年記念とかいって「FuzzyBASICコンパイラ」に「ZEDA-3」に「MML」だって？　こんなときにこんな嬉しい記事が載るなんて！　早く試験よ終われ！
丹羽　章暢（18）愛知県

◆「Oh！MZその筋事典」，これを見たとたん，背筋をなにかが走った。ああ，今日もまた，「その筋」を目指す男たちが，この事典を見ながら「その筋」しているのであろうか……。
西村　直樹（16）兵庫県

◆Oh！MZは“その筋”とともに歩み続けてきたように思います，「その筋事典」を見るかぎり。6月号もかなりの「その筋」があってなかなか読みごたえがありました。これからも，Oh！MZにはより“その筋”してほしいと思います。“その筋”のないOh！MZはOh！MZにあらず！
中山　貴弘（17）千葉県

◆特別企画の「Oh！MZその筋事典」はよかった。とても「その筋」の数を数える気にはなれなかったが，最初のページだけ数えたみた。28個。しかしそのページには「27」と書いてあった。うーん，その筋である。　戸嶋　秀和（16）大阪府

◆5周年おめでとうございます。Oh！MZは半分くらいしか持っていない私です。最初にOh！MZを買ったときはパソコン初心者で，あの「試験に出るX1」を読み，情報処理の国家試験ではX1を使っているのかと本気で考えていました。あのころに比べると，私も成長したものだなあ。
浅田　政人（16）愛知県

◆そうですか，もう5年ですか。その間に世の中ずいぶん変わってしまいましたね。長いような，そうでもないような……。ああ，戻れるものならあのころに戻りたい。けどパソコン界のレベルはもちろん戻ってほしくない。
上村　信晃（19）東京都

田村　憲生（18）鳥取県

◆X68000用のZ80Hボードを作り，そのS-OSを開発してほしい。それからシャープにお願いがある。NECのディスプレイテレビC-21M729のような大型ディスプレイを開発してほしい。スペックとしては，3モードオートスキャンはCZ-600Dと同等以上でNTSC信号のときオーバースキャンを最小限にして有効画面をいっぱいまで表示すること。Y/C分離の入出力端子を入力3系統，出力1系統付ける。21ピンマルチ入力2，出力1，ビデオ入力4，出力2。15ピンRGB入力2，8ピンRGB入力2。スピーカー出力端子，ステレオ2系統，サラウンドもついでに付けて。マスクピッチは0.4，画面サイズ25インチくらい。ノッチフィルタ，速度変調，くし形フィルタなども。水平解像度600本以上，チルトスタンド，オートターン，倍密度ノンインタレース変換，EDTV対応チューナ，文字多重チューナ付き，もちろんX68000から操作できるようにして。スーパーインポーズももちろんできること。で，価格は25万円くらいで！
古澤　千年（27）兵庫県

ついでに音声多重放送を受信できるともっといいんだけどなあ。

◆STUDIO MZを読んでいると，アンビバレンスという言葉が出てきた。なんとなく聞いたことはあったのであるがきちんと意味を把握していなかったので，いいチャンスと思い辞書をひいてみる。ambivalence，なるほどそういう意味だったのか。うーん奥が深い。そういえば，3ページ前にもわからない言葉があった。満開製作所長さんのおっしゃるソノスジティである。辞書を繰りながらあれこれつづりを考えているうちに言葉を忘れてしまった。メモにとろうと，ソノス…まで書いたとき，私の脳で分割整理されている情報にその言葉が結びついた。そうか，「その筋-ty」だったのか。そうなんですよね？　加藤　正直（23）栃木県

そのとおりです。辞書にはないけど試験には出るかもしれませんね。

◆なんと，最近僕の友人が2人もX1turboを買いました。僕もとーってもほしいんだけど，いつ見ても「残高164円」と書かれている貯金通帳が情けない。　坂本　秋浩（15）兵庫県

ゼロから出発する人もいます。焦らずがんばりましょう。

◆私は愛機MZ-1500で，あるゲームを開発中です。

日が待ち遠しくなりました。私程度のビギナーにもわかる内容の記事をもう少しお願いします。
中葉　淳一（44）千葉県

初心者の方にも対応するよういろいろ考えています。今後もよろしくおつきあいください。

◆ヨブ記よりも面白い。伝道の書よりも面白い。シオンのプロトコルより面白い。五輪の書より面白い。ドラえもんより面白い。聖書読むより面白い。法華経より面白い。アニメージュより面白い。
高田　勝成（18）大阪府

それってOh！MZのこと？　おほめにあずかり恐縮です。

◆家庭菜園の茄子って本当に失敗作がないのですか？　確かに“BLOOM”には失敗作はありませんでしたけど……。おっと，間違えた，これはOh！MZのハガキだった。　井上　宗春（20）群馬県

そもそも家庭菜園では「失敗作」の定義がないんです。自分で作るとみな可愛くって。

◆「2001年宇宙の旅」にはまいった。なにを隠そうこのゲームは，私が初めて買った市販ゲームソフトなのだ。あの音，あの解説，すべてが懐かしい。ああ，あのころはよかった。ところで，このゲームに裏ワザがあったのを知っていますか。右下の安全地帯などでずっとしゃがんでいると……。
木下　孝雄（15）兵庫県

ゲームのプレイもプログラミングも同じ感覚で楽しんでいたころから，裏ワザってあったんですね。

◆小学校低学年生向けの数学の絵本（CRTに絵が出るもの）を作る記事を載せてくださーい。僕が原稿書きます。　西岡　孝昭（38）三重県

パソコン紙芝居したりして。いまどきの小学生って恵まれてるんだな。

◆新作VG「ミッドナイト・ランディング」，これはなかなか面白い。「アウトラン」とともにぜひX68000に移植してほしいものです。出来のよいソフトがなによりもハードの販売実績に反映されるというのは，PC-98で実証されているのだ。ソフトハウス様，ぜひがんばってください。
鈴木　理雄（19）東京都

聞こえましたか，ソフトハウスさん？　のんびりしていられませんよ。

完成したらOh！MZに最初に送ります。S-OSは素晴しい企画なので永遠に続けてください。NEW BEMSも期待しています。共通化万歳！
大曽根　真一（20）茨城県

パソコンユーザーにとっての醍醐味はやはりオリジナルプログラムの作成。開発ツールがそこで役立てれば本望です。

◆パソコンに向かっていると時間のたつのを忘れます。仕事で帰りが遅いため触れる時間が少なくて残念。Oh！MZの記事はいつもとても面白く，毎月楽しみにしています。今井　清美（39）奈良県

時間を忘れて熱中できるってすてきですね。これからもそのお手伝いをさせてください。

◆待望のビデオプリンタが周辺機器の期待の新人（新機かな？）として発売されたのはたいへん喜ばしいことです。ぜひ欲しいものですが……。
五島　洋一（19）神奈川県

パソコングラフィックの夢を広げてくれるビデオプリンタ。問題は先立つものですか？

◆「仕事に必要だ」と大見得切ってX1turboを買い，“べえしっく”もロクにわからないまま妻の目を盗んで「めぞん一刻」に興じる。もうすぐ40だというのに情けない。若いモンにはかなわんと思いつつ読むOh！MZ。それでも2年前の記事からようやく理解し始め，ひとりでニンマリ。やはり辛抱ですね。これからもヨロシク。
三田　和英（38）千葉県

「若いモンにはかなわん」とおっしゃる人は多いけど，「いちばんよかった記事」など拝見すると，どなたもなかなか「その筋」ですよ。

◆2歳になるうちの息子はパソコンのことを「バンバン」と呼ぶ。我が愛機が毎度どんな目にあっているか想像できるであろう……。MZ-2500よ，おまえもよくよく運のないヤツだ。
秋葉　政利（26）茨城県

それでも健気に動いてるんですか？　さすがはヘビーデューティMZ！

◆暑い夏に身体を休めるため夏休みはある（ましてや沖縄）。しかし，ワープロ代わりにX1とソフトとプリンタが欲しいといったらバイトでかせげといわれた。夏休みがバイトでつぶれる――（ましてや沖縄）。　伊舎堂　盛行（15）沖縄県

秋にはX1があなたの机の上でニッコリ笑っているところを想像して。ほーら，やる気が出てきたでしょ。

ぼくらの掲示板

●掲載ご希望の方は，官製ハガキに項目（売る・買う・氏名・年齢・連絡方法……）を明記してお申し込みください。
●ソフトの売買，交換については，いっさい掲載できません。
●取り引きについては当編集室では責任を負いかねます。
●応募者多数の場合，掲載できない場合もあります。

仲　間

★「CANDY Network」ではパソコン通信の会員を募集しています。300ボー，パリティなし，データ長8ビット，ストップビット1，XON/XOFFあり，SI/SOなし。運営時間は午前0時から午前6時（夜半から早朝）までです。友好クラブのメンバーも募集し，BBS上に友好クラブ専用のBOXを作る予定もあります。詳しくは下記へご連絡ください。〒206　東京都多摩市永山5-1-9　鳥羽隆史（18）

★MZ-1500ユーザーの皆さん，「JMC1500」では第3期会員を募集しています。会報発行や情報交換などいろいろ活動中。詳しくは60円切手を同封のうえ連絡を。〒143　東京都大田区中央3-9-13　吉野信義（15）

★当サークル「坂本組」では，毎月の会報発行やゲームの情報交換を中心に活動中。できたばかりで会員はまだ10名ほどですが，ぜひ仲間になってください。対象機種はX1です。詳しくは60円切手を同封して連絡を。〒037-03　青森県北津軽郡中里町富野千歳311-1　坂本信臣（15）

★X1のテープ版またはX1turboZのユーザーの方，情報交換しましょう。とくにテープユーザーとしては，ディスクがなくてもまだまだ現役のX1でがんばりたいと思います。連絡ください。〒371　群馬県前橋市本町2-9-9　相沢忠之（16）

★MZ-2500のユーザーズクラブ「SMDCC」の仲間になりませんか，メンバーが全員参加して作られるディスク会報を発行しています。詳しくは60円切手を貼った返信用封筒を同封のうえ，Oh！MZ 9月号の発売日ごろまでに連絡ください。〒252　神奈川県綾瀬市深谷4595-6　増田耕一（19）

★冬コミに向けて初めて同人誌を作るため，オリジナリティのある執筆者募集。ジャンルは問いません。SF，ファンタジーに興味のある人歓迎します。自己PRを付けてOB.MZ係へ手紙ください。〒222　神奈川県横浜市港北区篠原町1531　大村邦嘉（15）

売ります

★フロッピーデイスクドライブCZ-502Fと拡張I/OポートCZ-8EPを送料込み5万6千円で。バラ売り可。往復ハガキで連絡をください。〒799-26　愛媛県松山市太山寺町2384-41-17　芳野聖吾（18）

★MZ-700/1500用拡張ボックスMZ-1U08およびドットプリンタMZ-1P14（ケーブル，トラクタフィード付き）を3万円で。往復ハガキにて連絡を。〒565　大阪府吹田市山田南29-5-207　権田考範（20）

★MZ-2200用フロッピーディスクのセット（ニデコNH-200F-MX）をボード，ケーブルその他を付けて6万円で（新品同様）。またプリンタMZ-1P07（ボード，ケーブル付き）を3万円以下で。〒250　神奈川県小田原市久野193-1-105　高島　泉（30）

★パーソナルテロッパCZ-8DT2を2万5千円で。往復ハガキで連絡ください。〒321-11　栃木県今市市明神280-5　増渕克美（28）

★X1シリーズ用ディスプレイテレビCZ-801Dを4万円くらいで。色はシルバー。完動，キズなし，付属品一式とテレビフィルタ付き。往復ハガキに希望価格を書いて連絡を。送料はこちらで負担します。〒030　青森県青森市大野若宮130-39　小野義之（19）

★漢字プリンタCZ-80PK（シルバー）を2万5千円くらいで。マニュアルあり，箱なし。往復ハガキで連絡を。〒973　福島県いわき市内郷宮町金坂6-12　前田英明（19）

★X1，MZ-2500用のツクモオリジナルフロッピーディスクTS-FDを2万円から2万5千円で。I/Fは付きません。往復ハガキにて連絡ください。〒331　埼玉県大宮市湯木町1-38-1　岩崎忠雄（18）

★フロッピーディスクCZ-503Fを送料別2万5千円で。今年3月に購入。インタフェイス，箱，付属品一式付き。キズなし。連絡は往復ハガキで。〒807　福岡県北九州市八幡西区浅川台1-14-7　中山修一（18）

★X1F用増設ドライブCZ-52FEを2万円で。未使用。ハガキにて連絡を。〒742-04　山口県玖珂郡周東町東午王ノ内　田中力（20）

★データレコーダCZ-8RL1を1万円で。付属品一式付き。往復ハガキにて連絡ください。〒051　北海道室蘭市新富町1-6-6　渡辺知己（15）

★プリンタMZ-1P14を2万円でケーブル，マニュアル，箱付き。連絡は往復ハガキで。〒340　埼玉県八潮市大瀬838-6　谷田守弘（30）

買います

★MZ-1F07，またはMZ-1500で使用できるフロッピーディスクドライブをI/F，ケーブル付きで5万円以下で。往復ハガキにて連絡ください。〒921　石川県石川郡野々市町藤平田2-160　輝多荘210号　渡辺一矢（18）

★X1用ユニバーサルI/OカードCZ-8U1のマニュアル付き完動品を6千円で買います。送料は当方負担。往復ハガキにて連絡を。〒393　長野県諏訪郡下諏訪町南四王6146-14　鳥羽安暠（16）

★MZ-1500用データレコーダMZ-1T03を送料込み6千円で。付属品一式付き希望。完動品なら多少のキズは可。ハガキで連絡を。〒210　神奈川県川崎市川崎区宮前町7-5戸穏ビル4F　高橋幸男（13）

バックナンバー

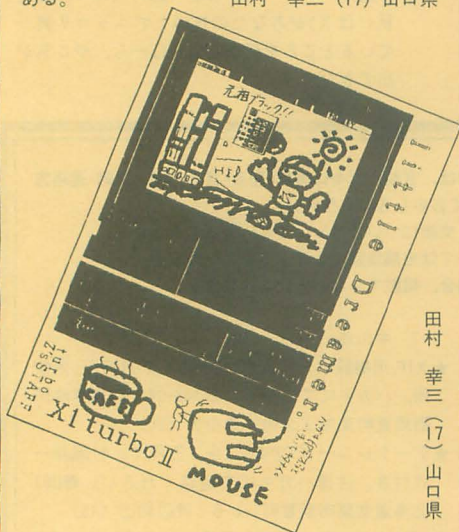
★1986年9月号を送料込み千円で。X1turbo用MAGICの記事が見たいのです。ハガキで連絡を。〒706-03　岡山県玉野市槌岡532　鳥越英司（18）

★1986年1月号から5月号を送料込み各千円で。全部まとめてなら6千円で買います。切り抜きは不可。往復ハガキで連絡ください。〒577　大阪府東大阪市菱屋東478さくら荘　葛目明久（18）

★1986年4月号から7月号までを送料込み2千円で。切り抜き不可。往復ハガキで連絡を。〒992　山形県米沢市林泉寺1-2-31　小川和夫（20）

その筋プレゼント 当選者発表

6月号で募集した愛読者プレゼント番外編に多数のご応募ありがとうございました。さっそく，当選者の方々のユニークなメッセージを紹介しましょう。さすがの編集室もたじたじのパワーです。

▶「Oh！MZ その筋事典」に“ブラックturbo”がないとはなにごとだ！　私はturboⅡが出てすぐ買ったのである。当時まだブラックは限定仕様だった。なのにいまではX1G，turboⅢ，Zまでブラックだ。やはり元祖ブラックのturboⅡは永遠に不滅なのである。　田村　幸三（17）山口県

田村　幸三（17）山口県

▶「祝一平・その筋テレホンカード」を作ってプレゼントまたは販売してほしい。NTTのもうけになるのはくやしいが。　山本　利彦（18）大阪府

▶その筋の見分け方：Oh！MZを読んだあと，表紙に筋ができる人はその筋，できない人はただの人。したがって6月号p.82の1984年6月号とp.83の1985年6月号を読んだ人はその筋に違いない。ちなみに僕はただの人。　奥山　徳二（18）愛知県

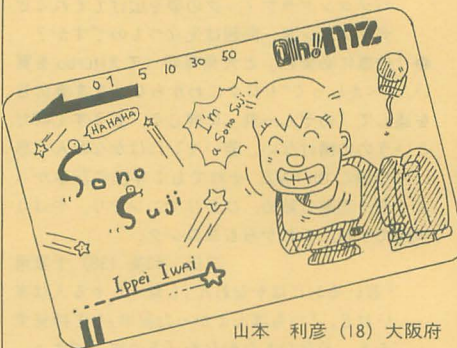
▶X1turboZを使っています。ところがお金がなくてCZ-600Dが買えず，CZ-801Dを使っています。早くほしいっ。　藪田　俊平（16）和歌山県

▶わが校の教えは「質実剛健」である。　中村　進悟（18）兵庫県

▶祝一平氏と清水和人氏ではどちらが強いのですか？　川田　秀一（18）埼玉県

▶ふと考えたんですが，郵便屋さんは，この宛先「その筋プレゼント係」というのを見ていったいなんだと思うんでしょうね。　斎藤　貴（21）岩手県

▶X1turboZの記事を読んだとき思った。Ⅲの次がZなら“スケバン刑事Ⅲ”の次は“スケバン刑事Z”なのだろうか。……話がそれてしまったが「Oh！MZその筋事典」は懐かしいものばかりで楽しかったなあ。　山崎　哲也（16）長野県

▶tinyXEVIOUSで6876の35 C2を00 C3にすると無敵になり，50EAの04に好きな数（$1C以上はよしたほうがいい）を入れるとスタート時の自機数の設定ができる。SPACE BLUSTER FZでも同じようなことがやれるけど，これじゃあB級情報。どの筋だろう？　石黒　博史（24）千葉県

山本　利彦（18）大阪府

▶ソフトハウスもしくは誰かにクトゥルフの呼び声の「テキストRPG」を作ってほしい。これはグラフィックで絵を描いてはだめだ。怪物の形が限定されてしまい，想像力による怪物というものが存在しなくなる。正体のわからない怖さというのもなかなかよい。　渡辺　一矢（18）石川県

▶オオカミは，顔の各部の毛を立てたり寝かせたりして表情を作り，感情を伝えます。みんなもマネしよう。　山田　昇治（18）徳島県

▶「その筋のご意見，ご希望」とだけ書いてくるのは何人いるだろうか。　山森　一人（18）愛知県

▶「電子ブロック」という超あの筋からパソコンの世界に首を突っ込んで4年と9カ月。当時最新だったMZ-1200はあっという間に旧世代の遺物となりがっくりしていたところへ，やたらと盛り上がっている雑誌がありました。Oh！MZ……なんてクサい響きなんだ。初めは過激に感じた文体にも慣れ，S-OSが毎月楽しみになった。　箕浦　真　岐阜県

▶神保町近くの予備校に通わなくなったいまでも，発売2日前にOh！MZを手にする優越感が忘れられず，買いに行ってます。　江副　滋（18）東京都

▶マニュアルも見ずNEW BASICを立ち上げZEDA-3を入力し始めた。途中，ハッと気づく。マシン語モニタでは……！　菅松　敏紀（18）宮城県

▶はっきりいってシャープは祝氏のアイデアをわずかながら採り入れていると思う。　望月　洋紀（17）山梨県

▶いまや「なんじゃこれは？」というマシンが多すぎる。「コンピュータは日進月歩である」なんて大ウソの売上税だ。　神原　洋司（30）広島県

▶「その筋事典」に1500/700 USER'S BULLETINがなかったのは残念。彼らは十分その筋です。あんな誰にもできないだろう製作記事を堂々と載せたのだから。　松永　和久　熊本県

▶パソコンマニアやアニメファンももちろんだが，鉄道ファンにも「その筋」な点がいっぱいある。　坂本　繁海（18）東京都

▶MZ-700には，256KバイトのC-MOS・S-RAMディスクと80桁ボードがあればPC-8801を超えることができるポテンシャルが秘められている！　だから作ってちょーだい。　塩月　達也（16）大分県

▶先日，あるキャバレーの前の看板に「その筋の人はご遠慮ください」と出ていた。「ほほう，祝さんも偉大になったもんだな」と思いました。次には“その筋キーホルダー”を店の前でちらつかせようと思います。　須藤　義一　東京都

▶私は故岡田有希子さんと南野陽子さんのファンである。しかし，5月号の好きなタレントベスト10を見て悲しくなった。なぜ斉藤由貴が2位なんだ，なぜ段田男や原田知世がいるんだ！　谷村新司はシブイが。X1シリーズのCMに南野さんを使うようにシャープの宣伝部に圧力をかけてください。大丈夫，落合やホーナーのようにきっと「もと」を取らせてくれると思う。　細井　太藏（19）愛知県

▶現在，私のパソコンMZ-731「政宗」は，拡張I/OボックスにZ8000が入っていて無敵になっている。本体のZ80は単に入出力用である。S-OSにも全機種共通16ビットボードなるものを発表してそれを育てていってはいかがだろう。　安達　英夫（17）青森県

▶Oh！MZ編集室を拝みたいのですが，途中に皇居があって家ではできません（右翼だと思われる）。だからよそへ引っ越していただけませんか？　犬飼　直彦　神奈川県

▶表紙がまたオークスターにならないかと思い，彼女の行方を追ってバックナンバーを開いた。いた，いた。1983年6月号に。ふむふむ。「おねえさんはお星様になったのよ」。さっそく望遠鏡をのぞく私であった。　石橋　一義（22）北海道

▶私の大学にもドラゴンを発見しました。基礎電子回路Ⅰの教授です。彼はまったくテキストとはかけ離れた授業をし，複素数の計算に触れるやイキナリ「次の時間までにこの計算をやってくること。みんなポケコン買っただろ。それでプログラムを組めば複素数の計算ができるから，各自工夫するように」という60000°Kの炎を吐くのです。誰か私にこのドラゴンを倒すアイテムをくれー。　本田　悦朗　宮崎県

奥山　徳二（18）愛知県

大野　真実（25）静岡県

▶満開２号には，古語「SP-1002」で書かれた豊富な古典を利用するための「MZ-80Kエミュレータ満開/金田一京助」を付属させてください。また，MZ-80K/C用「満開２号コンパチボード　満開/情」を発売してください。それから，満開２号の内部伝達信号にタキオンを利用する計画はないのでしょうか。また，祝氏は本当にゾウが踏んでも壊れないのですか？　　熊谷　聡（15）兵庫県

▶あれはまだ私が純真無垢であった２年と６カ月前のことである。変人として有名な友人宅へ遊びに行くと，本棚にOh!MZはあった。開くと内輪受けとしか思えない単語が並ぶ。そのときは「なんという訳のわからん本ぢゃ！」と半分怒っていた。やがて時は流れ，気づいたときはもう遅かった。「遺言級」の言葉に魅せられ，いまではシャープなしでは禁断症状に陥ってしまう。こんな身体にした「その筋」が憎い。　藪本　輝夫（18）大阪府

▶「その筋」てのは表面に出さずに「かくし味」のようにするべきだ。それがなんだよ！「その筋事典」まで出してしまってさ。祝さんや清水さんの文章の中にぽろっと出てきて，わかる人にはわかるというマニアチックなところがよかったのに。
溝口　伸一（18）愛知県

▶満開６号には８進プロセッサを，そして７号には究極の10進プロセッサを使ってはいかがでしょう。名称は６号がタコ八郎，７号が十露盤くんです。　　西村　進（17）石川県

▶全快１号は“ぜんまいちゃん”などというミーハーな機械も悪いとはいわぬが，やはり戦う読者のために戦闘モードを付けたキカイダーや仮面ライダーなどのバージョンがほしい。３体セットでお得なアクマイザー３バージョンも。ちなみに戦闘モードに移る場合，“変身”しなければならないのである。　　中山　典英（18）福岡県

▶剣をキーボードに持ち換え，あるいはスペルブックをマニュアルに換えて戦い続ける我々に召集がかけられた。Oh!MZというドラゴンを倒せ！と。倒したらヤバい気もするが。それに応えるべく，僕は「その筋」と彫り込まれたキーホルダーをカバンに付けて仲間を見つけなければならないのだ。　田中　真一（16）三重県

▶ものごとを受け持ち取り締まる官庁，当局。その向き（の人）。警察。やくざ関係。辞書に出てるのは以上ですが，「その筋」はこのうちどれなのですか？
鈴木　裕介（14）神奈川県

▶ある日私は決心した。自分に合ったDOS，エディタ，アセンブラを自分で作る。DOSはたいへんそうなので目標をDOS内蔵のモニタに定めて作った。ところがIOCSを含めて11Kバイト近くになってしまった。どうせならマルチウインドウにしようとしたのが命取りで，メモリを喰うこと。こんな私はその筋でしょうか？　　足立　勇（17）愛知県

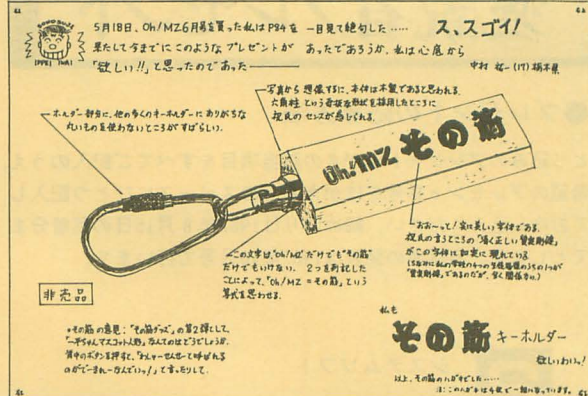
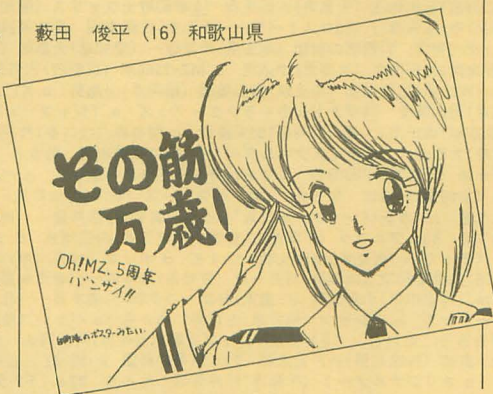
▶こんなことが現代の民主主義社会の日本にあっていいのか？　Oh!MZ1987年６月号「その筋キーホルダー50名」。しかし，ひとつの単語が欠けていた。「抽選」。これはOh!MZの陰謀だ。読者の実力でキーホルダーを勝ち取れ，という。なんてひどい仕打ちだ。極悪非道，ゴルゴ13級ドラゴンと変貌してしまったのか。親の遺言はゲームの世界だけではなかった。それはOh!MZそのものだったのだ！　そして読者は，倒せるはずのないドラゴンと千日手でにらみ合うのだった……。
東野　世士宏（18）奈良県

▶かのデーモン小暮が祝さんと同じような話し方をしているのですが，祝さんは悪魔教の信者か，それともデーモン小暮のほうが祝さんの影響を受けているのだろうか？　沖本　健治（19）大阪府

▶友人によると，月刊「OUT」で使われる用語が登場するとその雑誌は滅亡するそうですが，大丈夫ですか？　　濱岡　裕樹　京都府

▶FuzzyBASICでなんとかカラーとブリンクをさせようと，0026Hにデータを書き込んだらできたのですが……。　　黒川　勝則（17）福井県

▶その筋キーホルダーは素晴しいデザインだ。さすがは祝先輩，その筋代表として木目を選ぶとはにくいねえ，この色男！　岡林　直樹（17）福井県

藪田　俊平（16）和歌山県

中村　祐一（17）栃木県

▶S-OSに知性と感性を装備する。たとえばどうしようもないタコなプログラムを書くと，親切にお手本プログラムを示してくれたり，励ましてくれたり。心が和むよ。　　坪田　吉和（20）石川県

▶友人に「俺なら本屋でOh!MZをレジに持っていく途中，わざとその筋キーホルダーを落とし，“ああ祝さんお許しください”と大声で叫ぶことができるぜ！」と言いそうになったが抑えた。だって彼は必死に打ち込んでいるSWORDを僕にもコピーさせてくれるのだから。柿沼　智幸（16）北海道

▶グラディウスでレーザーを取ったら「ミーン，ミーン」と言い，ビッグバイパーがやられたら「ツキャン，ツキャン」と言いましょう。以後，ツキャンするのいうのは自機がやられることを指すようになったのでした。　　吹田　幸介（18）秋田県

▶祝一平vs.高橋雄一の特集を組んでは。
小川　祐司（20）栃木県

▶その筋はもともと「試験に出るX1」のワクにあった筋のことだったのだ。最近はその「筋」を見かけないなあ。　　長田　純也（17）岡山県

▶近ごろ私はぜんまいちゃんに不満です。アプリケーションソフトが出ないからだ。いっそ満開２号に載っけたら？　　田中　一成（15）京都府

▶私は発見した！「その筋事典」のバグだ。1984年５月号の27ページにFD-700の説明がある。その中に燦然と「LOAD“CMT：KYOKO”，A」の文字が輝いているだろう！　響子さんが出たのは11月号が最初というのは間違いだ。やった！　ついに俺はOh!MZの揚げ足を取ることができた！
大道　亮（15）埼玉県

▶なぜ「Oh!MZその筋事典」に「VF-1S STRIKE VALKYRIE」が載ってないんだ！　1985年１月号では愛読者プレゼントになっているのに。
戸嶋　秀和（16）大阪府

▶理想のパソコンは64ビット３次元構造CPUを搭載し，内蔵の読み書き可能な光ディスクによりカラオケもできるので「宴会１号」と呼ばれる。装備されるソフトは「幹事くん」である。
藤田　真史（19）北海道

以上の方にキーホルダーを送らせていただきます。発送は遅れることもありますのでご了承ください。

愛読者プレゼント

●プレゼントの応募方法

とじ込みのアンケートはがきの該当項目をすべてご記入のうえ，希望のプレゼント番号をはがき右上のスペースにひとつ記入してお申し込みください。締め切りは1987年8月15日の到着分までとします。当選者の発表は1987年10月号で行います。

1

キティ・エンタープライズ
☎03(499)4271

扉を開けて 3名

X1/X1turbo用 5D版 7,800円

新井素子原作のアドベンチャーSFがRPGになった。シミュレーションやアドベンチャーの要素もふんだん。さあ，長い眠りから覚めた女王になって異世界の人々を救いに旅立とう。

2

システムソフト
☎092(521)0337

a)オーガ ポストカードセット 10名
b) ポスター 5名
c)大戦略II ポスター 5名

シミュレーションゲームとして人気のオーガのオリジナルポストカードとポスター，そしてX1シリーズ用の発売を期待しつつ大戦略IIのポスターを計20名にプレゼント。

3

サザンパシフィック
☎045(314)9514

オリジナルTシャツ 5名

夏らしいさわやかなプリントのサザンパシフィックオリジナルTシャツ。サイズはL。

4

シャープ
☎06(621)1221,
03(260)1161

X1/X1turboシリーズユーザーのための情報誌ソフトウェアフィールドを20名の方に。メーカー各社から発売されているアプリケーションソフト，周辺機器，関連書籍などの情報がいっぱい。

ソフトウェアフィールドvol.11 20名

5

日本ソフトバンク

オリジナルテレホンカード 10名

この秋に創刊されるコンピュータマガジン『THE COMPUTER』を記念してオリジナルテレホンカードを10名に。

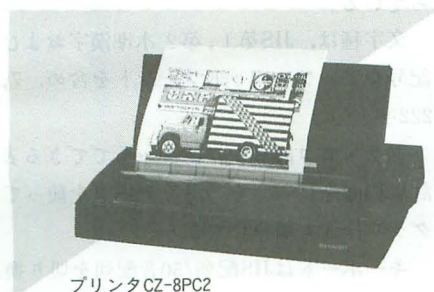
6月号 5周年記念特大 プレゼント当選者発表

[1]a.**電子手帳**（福岡県）秋根昌章（茨城県）野村正文（大阪府）西谷太郎 b.**ハンディコピー**（香川県）宮本幸展（東京都）秋山憲司（富山県）谷角正昭 [2]**A列車で行こう** a.X1turbo用（鹿児島県）吉丸太一（滋賀県）西山俊治（群馬県）阿部正一 b.MZ-2500用（埼玉県）石田貴嗣（大阪府）高倉和宏（東京都）市村泰子 [3]**ウイバーンテレホンカード**（北海道）山内和茂（東京都）柿崎章夫 箭内敬（福岡県）高口長三（岐阜県）佐々木満（石川県）中蔵征英（長崎県）谷口勝彦（栃木県）若井田和広（大阪府）西村泰之（和歌山県）堂脇尚子 [4]a.**ディーヴァ**（千葉県）秋葉貴男（兵庫県）栗田弘幸 b.**レイドック**（埼玉県）小林修 上野貴市 [5]**バッくんエプロン**（愛知県）関利昭 [6]**イミテーションシティ**（愛知県）山内崇義（北海道）水上秀博（東京都）内藤晋 [7]a.**D-SIDE**（静岡県）松浦聡志（東京都）高橋暁人 b.**フルーツ・フィールド**（長崎県）上田聖（香川県）矢野範晃 c.**走れ！ SKYLINE**（東京都）中沢博幸 高橋智津子 d.**クラックス**（神奈川県）磯崎行宏（埼玉県）常世田一郎（宮崎県）山口幸一 e.**ファイティングゲームズ**（大阪府）柴田方秀（埼玉県）畔見和俊（愛知県）関口健三 [8]a.**迷宮への扉**（神奈川県）高橋幸男 佐藤英夫（大阪府）亀山一久 b.**ソフィア**（愛知県）伊藤浩二（北海道）合田雅広（千葉県）吉田千春 [9]**夢幻戦士ヴァリス**（愛知県）辻井知文（秋田県）大渕茂視（京都府）額田恵介（山口県）弘永直行 [10]**ロストパワー**（千葉県）佐藤支利（愛知県）坂野弘幸（京都府）木全克徳（大阪府）龍野智哉（福岡県）武内邦博 [11]**覇邪の封印**（埼玉県）千葉啓一（愛知県）大島靖（長野県）牛山治信 [12]**殺人倶楽部** a.X1/X1turbo用（奈良県）住田浩之（兵庫県）中込浩 b.MZ-2500用（京都府）沢野基志（愛媛県）石川亨 [13]**ロボレス2001** a.MZ-2500用（岡山県）宮岡福樹（埼玉県）吉田周理（福岡県）小畠努 b.X1/X1turbo用（愛知県）大草幸一（山口県）中村謙二（沖縄県）野田廣史 [14]**ザナドゥ キャラクターグッズ** a.**Tシャツ**（大阪府）松村源史（愛知県）水野将徳 b.**エプロン赤／**（東京都）畑田浩之 **黒／**（静岡県）野末勝樹 c.**財布赤／**（山口県）片平明徳（岡山県）寺尾文治 **黒／**（千葉県）柴崎健次郎（大阪府）大西健太 d.**ディスクホルダー**（宮城県）高橋吉春 他9名 e.**バッジ**（愛知県）中村隆一 他3名 f.**カンペンケース**（千葉県）岡田博明 **ロマンシアキャラクターグッズ** g.**バッジ**（京都府）森川正寿 他7名 h.**ディスクホルダー**（宮崎県）川崎修 他9名 i.**カンペンケース**（新潟県）山口徹 j.**太陽の神殿ディスクホルダー**（広島県）谷岡隆浩 他9名 k.**“ガンバレ・ファルコム”バッジ**（青森県）根来健一 他99名 l.**情報誌「ファルコム」**（香川県）藤野久也 他9名 [15]a.**アルファ**（千葉県）久重崇史（埼玉県）内田博章 b.**ブラスティー**（千葉県）中村隆司（長野県）有賀正訓 c.**ステッカー**（愛知県）加藤文明 他4名 d.**ポップ**（神奈川県）井上忠行 他3名 [16]**森田和郎の将棋8ビット版**（埼玉県）高橋奈加次（山梨県）石井一成（愛知県）岩本健一（鹿児島県）小倉輝久（富山県）笠島秀樹 [17]**JETターボターミナル**（神奈川県）小森武雄（三重県）林雄一（京都府）橋本晋一（北海道）高波秀幸（大阪府）内川達夫 [18]**スーパーランボーグッズ** a.**Tシャツ**（福井県）中哲男 他4名 b.**バッジ**（鳥取県）大隈清治 他9名 c.**ビニールバッグ**（青森県）神谷敏子 他11名 [19]**トップルジップ下敷**（宮崎県）町元真也 他4名 [20]**はーりぃふぉっくすTシャツ**（京都府）橋本義雄（神奈川県）折戸光太郎 [21]a.**信長の野望 全・国・版**（岩手県）菊池隆之（山口県）河上忍和（長野県）小林由未 b.**オリジナルノート**（千葉県）松井芳昭 他4名 [22]a.**F2グランプリ**（福岡県）日浦寛二 b.**大脱走**（群馬県）清水哲郎 c.**ハイドライド**（東京都）小岩寿之 d.**ノート**（京都府）内藤宏人 他9名 [23]a.**照魔鏡の伝説**（岩手県）舘沢和貴（北海道）福嶋憲 b.**トランシルバニアII**（長野県）水上淳一（東京都）田村義博 c.**マスカレード**（宮城県）佐藤明彦（大阪府）森山巧 d.**ラスベガス**（三重県）伊藤俊宏（岩手県）高橋克公 e.**アマゾネス**（北海道）山下隆之（茨城県）佐藤好和 [24]**ブラックオニキスファイルケース**（島根県）中島基 他4名 [25]a.**『実録！天才プログラマー』**（宮城県）細川晃 他4名 b.**ペンケース**（東京都）森山伸行 他4名（以上敬称略）

以上の方々が当選されました。おめでとうございます。なお，賞品は順次発送いたしますが，入荷の状況によって多少遅れる場合もありますのであらかじめご了承ください。

ペンギン情報コーナー

●NEW PRODUCTS

24ドット熱転写カラー漢字プリンタ
CZ-8PC2
シャープ

プリンタCZ-8PC2

シャープは，24ドットの熱転写カラー漢字プリンタCZ-8PC1の後継機として，X1/X1turboシリーズおよびX68000用プリンタCZ-8PC2を７月から発売する。CZ-8PC1ではオプションだったJIS第２水準漢字ROMが標準装備されて，価格は変わらず69,800円。

CZ-8PC2は，文字構成24×24ドットの熱転写/感熱方式で，カラーリボンCZ-8PC1-2（リフィル用800円）を使えばカラー印字が可能。X1turboZのアナログRGBの画像データなども，細かな階調でハードコピーにとることができる。黒色リボンCZ-8PC1-1は700円。

プリント用紙のサイズは，B4判縦からB5判横のカット紙のほか，はがき印字も可能。用紙のセットはセミオートローディング方式なので，１枚ずつ給紙する必要がある。

印字速度は，漢字の場合で１秒間30文字。一行最大印字数は，漢字で51文字，縮小のANK文字で136文字。

〈問い合わせ先〉
シャープ㈱　☎06(621)1221, 03(260)1161

MZ-2861用1MバイトRAMボードほか

シャープからMZ-2861用の1MバイトRAMボード(MZ-1R35 55,000円)，同ボード用増設RAMカード(MZ-1R36 45,000円)，80287 数値演算コプロセッサ(MZ-1M12 90,000円)，およびMZ-2500モード用拡張ユニット(MZ-1U09 9,000円)が発売された。

増設RAMは現在のMS-DOSが80286対応となっていないため，もっぱらRAMディスクとして用いられる。本体内に6Mバイト(FD5枚分)まで拡張が可能だが，IPLスイッチを押すと初期化されてしまうので注意が必要である。

24ドット熱転写カラー漢字プリンタ
エプソン AP-80EX
セイコーエプソン

セイコーエプソンは，24×24ドット熱転写カラー漢字プリンタAP-80Kの後継機としてAP-80EX(64,800円）を６月15日より発売開始した。X1/X1turboシリーズ，X68000，およびMZ-2500シリーズにて使用できる。

漢字は24×24ドットの明朝体で，印字速度は53文字/秒，半角のANK文字は80字/秒。また，JIS第２水準漢字ROMを標準装備している。

用紙サイズはカットシートフィーダ（オプションで13,000円)を使った場合，幅140mmから216mmのカット紙が使用でき，ロール紙では210mmから216mm。桁数は普通文字で80桁，縮小文字では144桁。

リボンカートリッジ(リフィル用１本800円）はテープの両面が使用でき，最大で約6.5万字の印字が可能。カラーリボンカートリッジ(1,000円）は，１本で７色印字を行える。

〈問い合わせ先〉
セイコーエプソン（株）　☎0266(52)3131

プリンタAP-80EX

マウスとジョイカード
CZ-8NM2/CZ-8NJ1
シャープ

X1/X1turboシリーズ用のマウスCZ-8NM2(6,800円)が発売された。これは，X1シリーズに接続する場合，別売のRS-232CマウスボードCZ-8BM2(19,800円)が必要になる。このマウスはX68000, MZ-2500でも使える。

また，X1/X1turboシリーズおよびX68000用のジョイカードCZ-8NJ1(1,700円）が７月から発売される。

〈問い合わせ先〉
シャープ㈱　☎06(621)1221, 03(260)1161

1200bps低価格モデム
MD-1200E
立石電機

立石電機は，データ通信機器 COM MATEシリーズの新製品として，通信速度1200/300bpsで24,800円という低価格のインテリジェントモデムMD-1200Eを６月から発売した。

制御コマンドは，米国ヘイズ社のATコマンドおよびCCITT規格のV.25bisの両方に準拠しており，どちらでも選択できる。通信プロトコルはCCITT V.21/22, BELL103/212Aに対応。選択信号はトーン(プッシュ回線）とパルス（ダイヤル回線）のいずれも可能になっている。

自動発着信や，相手側の通信速度を認識してそれに合わせる自動設定機能もサポートしている。また，アボート・タイマーを

モデムMD-1200E

設定しておくと，データのやりとりが5分間ない場合，自動的に回線が切れるようにセットできるので，回線の切り忘れによるロスを防ぐことができる。また，送受話器を取らずにパソコンから自動的に電話発信が可能な，オンフックダイヤル機能も搭載しているので，多くの相手に電話しなければならないときなどに便利。

〈問い合わせ先〉
立石電機㈱　☎03(436)7233

モデム2機種
TCOM1200B/2400C
タムラ製作所

タムラ製作所は，1200bps全二重モデムTCOM1200B(34,800円)と2400bps半二重モデムTCOM2400(70,000円)を6月から発売した。

TCOM1200Bは，CCITT V.25bis またはATコマンドが使用でき，CCITT V.21/22とBell103/212Aのいずれでも選択できる。自動発着信，通信/通話切り換えが可能，ダイヤルパルスとトーンダイヤルのいずれにも対応している。

モデムTCOM2400C

TCOM2400CはCCITT V.25bis準拠のBSCまたはHDLCコマンドを使用してCCITT V.26bisに対応している。

〈問い合わせ先〉
㈱タムラ製作所　☎03(978)2090

パーソナルワープロ
エプソン ワードバンク-LXT
セイコーエプソン

パーソナルワープロの新製品ワードバンク-LXT(128,000円)は，最大200字までを一括変換でき，固有名詞や複合語を含む約13万語の辞書を備え，変換効率の向上を図っている。

文字種は，JIS第1，第2水準漢字および記号やアルファベットフォントを含め，7,222字。

タテ・ヨコ計算がワンタッチでできる表計算機能を持ち，その計算データを使ってグラフ作成も簡単にできる。

キーボードはJIS配列/50音配列を切り換えでき，プリンタ部は熱転写・感熱式で全角文字の構成は24×24ドット，3.5インチFDDを1基内蔵している。用紙サイズはハガ

1987-08
外資勢に
注目

Again Watch

大手外資メーカーの動向

3大コンピュータメーカーといえばIBM，ユニシス，DECだが，最近この3社の日本をめぐる動きが激しくなっている。

まずIBM。今年創立50周年を迎えた日本アイ・ビー・エムの記念式典に出席するため，本社のエイカーズ会長が来日した。さきごろ米国以外の地域オペレーションを再編し，「そろそろ日本に対する施策変更の時期」といわれているタイミングであるだけに，業界の噂も「椎名社長の退陣か」という説が出るところまでエスカレート。このような噂の根拠としては昨年12月期決算で初の減収減益決算(売上高8785億円，前期比3.9%減)となったことがあげられる。このため，電算機部門の売上高は日本電気に抜かれて3位に転落(トップは富士通)し，4位の日立製作所との間も縮まっているからだ。この成績に，米国本社が満足するはずはなく，やはり新しいオペレーションはあるかもしれない。現実にパーソナルシステム/55ではパソコンで初めて米国と同じ製品を日本で売る。変わることだけは確かだ。

次にユニシスだが，同社の日本における子会社は日本ユニバックとバロースの2社である。昨年7月に米国スペリー社と米国バロース社が合併してユニシスが誕生したいきさつから，日本の両社も合併が憶測されているところだ。しかし日本ユニバックだけが上場しているため株価の関係から最低でも3カ月は準備期間が必要だそうで，そうなると早くても合併は今年の10月になる。合併問題については日本ユニバックの大株主，三井物産と米国ユニシスとの間で交渉が続いているが，断が下されるのは時間の問題だろう。汎用機市場における対IBMの急先鋒としての強化策だったが，日本での新政策打ち出しは合併後になるだけにその時期が注目される。

最後にDEC。さきごろケネス・オルセン会長が来日したが，そのときの話によると，いよいよ日本でのVAXの現地生産(アセンブルのみ)を9月にも開始するようだ。また，この6月1日には子会社の日本DEC研究開発センターを吸収合併した。1989年には現在の東京・三番町研究所に加えて横浜研究所も開設するそうで，研究開発および部品調達にも力を入れる。

以上が3大メーカーの動向だが，いずれも日本での販売施策を非英語圏におけるマーケティングの柱として打ち出しているだけに，今後も一層の強化は確実であり，富士通，日電などの国産勢とのシェア争い激化は必至と見られる。

難しい“日本支社”運営

コンピュータランド・ジャパンが6月末に，カテナに買収された。コンピュータランドは36店のフランチャイズ店を日本国内に持っているが，うち9店をカテナが所有しており，もともと近い関係にはあった。このたび親会社の米国コンピュータランドから，カテナが全株式を取得したことにより，米国本社とコンピュータランド・ジャパンは資本関係がなくなったわけだが，従来どおりの業務は続けるという。

コンピュータランド・ジャパンは，1982年7月に米国コンピュータランドと兼松江

キからB4横まで。ディスプレイはSTN液晶で40字×10行表示。

また，ワードバンクシリーズにはオプション機能が豊富であり，エプソンイメージリーダ3(36,800円)やイメージリーダワイド(46,800円)により画像の中間調までを読み取れるほか，ハンディプリンタ(19,800円)でちょっとしたメッセージや名前などを印字したり，ラクラクボード(29,800円)で手書き文字の入力，グラフィックタブレット(19,800円)でイラストや図形の入力をしたりすることも可能だ。

さらに，プリンタインタフェイス(5,800円)を装着して24ピン漢字プリンタを接続したり，通信セット(15,000円)でモデムを介して電話回線につなげば，ネットワークにアクセスしたり，ワードバンク-LXTや他のパソコンとデータのやりとりもできる。

〈問い合わせ先〉

セイコーエプソン㈱　☎0266(52)3131

ワードバンク-LXT

音声合成ハード

VOICE-BOX

ログ

PCMによる音声合成出力が可能なボイスシンセサイザーVOICE-BOX(9,800円)が発売になった。セントロニクス準拠のインタフェイスを装備したパソコンならプリンタポートを介して接続可能。

さまざまな音をリアルに再生でき，人間の肉声などは，声の主が誰であるかの識別も可能な音質で出力できる。

また，X1／X1turbo用アドベンチャーゲームソフト「極楽鳥パラダイス」が付属しており，人の声をはじめとする効果音を臨場感たっぷりに楽しめる。サイズは，縦88×横68×奥行68㎜の手のひらに乗る大きさ。

〈問い合わせ先〉

㈱ログ　☎03(837)2595

VOICE-BOX

●INFORMATION

全国草の根BBS大会開催

日本全国に1000局ほど開局されているというBBSだが，このたび，ふだんはあまり交流のないこうしたBBS間の連絡を深め，運営上発生するさまざまな問題の討議などを行おうと「全国草の根BBS大会」が催されることになった。FORESIGHT,POPCOM-NETなど7BBSが協賛している。

日時は昭和62年8月23日(日曜日)の午後1時から4時半まで。会場は東京・科学技術館(入館料500円)の2階展示室。会費無料。

〈問い合わせ先〉

YAS-NET　アクセス番号03(943)9800

通信パラメータ　N81XNN

ゲストID：ADD0000，パスワード：0000

商が折半出資して設立したが，2年前に兼松江商が資本を引き揚げて，コンピュータランド社の100％子会社になった。このときも兼松江商とコンピュータランドとの間で経営に対する意見の対立があったようだ。

今回のケースでは，運営が手に負えなくなった米国本社と買収に積極的なカテナとの意見が合致したもの。グループ会社としての機能は残すことから全株式の委譲を決めたようだ。

昨年2月にアスキーとマイクロソフトが袂を分かったのは，米国ソフト会社とその日本法人との間がうまくいかないことを端的に表した事件だったが，その後も，昨年6月にアップルコンピュータ・ジャパンが社長交替し，11月にはマイクロプロ・ジャパンの社長が代わっている。マイクロフォーカス・ジャパンの経営陣が総入れ替えになったのも昨年だった。

日本の経営感覚では，まず土台づくりをしてそれから拡販攻勢に入る。しかし，米国のそれはいきなり大ヒットを求める。ここが決定的違いだ。しかも「世界でもっとも大きな国アメリカで成功しているのになぜ日本で売れないのか」という思いが米国側にはある。そのあたりのギャップを日本人スタッフがよほどうまく埋め，米国側経営陣の理解を得られない限り，米国本社と日本法人との関係は成功しないのだろう。

ロータスはいまのところ順風満帆のようだ。1年たったマイクロソフトも合格点をつけていい。しかしいつ米国本社が「無理な要求」を出してこないという保証はないのだ。

Short Again

CD-V発進

このほどキャニオンとポニーが「映像つきCD」のCD-Vソフトを発売した。ハード(プレーヤー)は6月1日にパイオニアが発売，続いて秋までには少なくとも松下とヤマハが発売する。いよいよCDで音楽を聞きながら映像を見られる時代になったわけだ。これでソニー・フィリップス連合のCDファミリーはCD，CD-ROM，CD-Iとあわせ，すべて商品化されたことになる。

セイコーエプソン

セイコーエプソンのPC-9801互換機であるPC-286モデル0が8月出荷分から改称し，BASIC-ROMつきのモデルも同時に発売される。ただしPC-286モデル0についてはこれまで，売れているという話を聞かない。

マイクロソフト版互換機

一部報道によると，マイクロソフトはIBM-PC/ATの日本語版互換機の仕様を決め，近くわが国のパソコンメーカーに商品化を呼びかける。詳細は不明だが，15社に呼びかけ，そのうち数社は年内に商品化しそう。強力なPC-9801対抗機になることは確実だ。

マスターネットが運営開始

明治乳業のパソコン通信サービス子会社であるマスターネットが，このたびモニタ会員3,000人に限定して実験サービスを開始した。JUST-PC2400bps高速通信が話題だが，本格的なサービスは今秋の予定。

Lattice Cが2万円を切る

ライフボートは，19万8,000円で販売中のCコンパイラ「Lattice C」の普及版として「Lattice Cパーソナル」を19,800円で7月18日に発売した。これは日本語処理を標準装備したサブセット商品。2万円を切りいよいよC言語は「ポストBASIC」になるのか。

(K.T.)

FILES Oh!MZ

このインデックスは，タイトル，注記――筆者名，誌名，月号，ページで構成されています。Z's STAFF PRO 68KやBASICなどの紹介記事が相次ぎ，目の離せないX68000。その影響もあって各マシンのユーザーは皆活気づいているようです。

参考書籍

I/O　工学社
ASCII　アスキー
THE BASIC　技術評論社
テクノポリス　徳間書店
Hacker　日本文芸社
パソコンワールド　コンピューターワールド・ジャパン
POPCOM　小学館
マイコン　電波新聞社
マイコンBASIC Magazine　電波新聞社
LOGIN　アスキー

新刊書案内

この本は雑誌「遊撃手」と「バグ・ニューズ」に掲載された対談，インタビュー集です。だから，「アンチ・ハッカー宣言」という書名はちょっと的外れです（少なくとも「宣言」などではない）。さて，肝心の内容ですが，なかなかに読みごたえがあるものになっています。考えてみれば不思議なことですが，「物の道理をわきまえたエンドユーザーの声」は，パソコンの世界ではあまり聞こえてこないようです。実際の大部分のユーザーは「マシンに過大な期待を寄せたりしない悟りきった技術者」でもなく，また，「自分の所有機種の欠点を指摘されると逆上してしまう視野の狭いマニア」でもないはずです。その点で本書は，「今まで不当に無視されていた人々の声」ともいえそうです。具体的な内容としては，「ウィザードリィ」や「ウルティマ」の作者たちへのインタビューや，作家，文化人のゲーム，ワープロをまな板に上げた対談などとなっています。特筆すべきは本書の脚注です。ほどよい「毒」が流れているにもかかわらず，「電脳都市」のように出しゃばったものになっていません。本書はいまいちばん望まれている「日常の物としてのコンピュータ」を指向しているといえます。　(Y.T)

アンチ・ハッカー宣言
「バグ・ニューズ」編ビー・エヌ・エヌ刊
A6判　200ページ　1,200円　☎03(238)1321

ネットワーキング&データコミュニケーション

コンピュータ・コミュニケーションに少しでも関係する人が必要とする知識を，基礎から解説したのが本書である。主としてビジネスユーザーを対象に，データ伝送の仕組みからインタフェイス，プロトコルなどについて述べ，またパソコン通信やLAN，ボイスメール，ビデオテックス，ISDNに至るまで幅広く触れている。さらに，こうしたデータ通信が発達していく中で起こるネットワークへの不法侵入や，それに対するセキュリティの重要性についても言及していて興味深い。

V.C.マーニー・ペティックス著　アスキー出版
A5判　264ページ　1,800円　☎03(486)1977

宇宙で食べるレタスの味

人類初の宇宙飛行士はソビエト連邦から誕生した。本書は，ジャーナリストとして同国を取材した著者による，ソ連宇宙開発事情のレポートである。宇宙ステーションという閉鎖された空間が人間にとってどんなものか，そこに芽生える文化はどんな可能性を持つのか。読むうちに，宇宙でレタスを食べるときの味を，あなたも知りたくなるかもしれない。著者はまた，なぜソ連がSDIに対しかくも神経質なのか，なぜ米国がSDI一本槍なのかも見えてきたようだ，と語っている。

若居亘著　同文書院
A6判　242ページ　1,300円☎03(359)9671

一般

MZ-80K/C/1200/700/1500

MZ-1500

▶画面数1000, 究極のパズルゲーム XOR
道の部分を制限歩数内で塗りつぶすというシンプルながら奥深いパズルゲーム。――編集部/岸和田高校物理部プロジェクトチーム, ASCII, 7月号, 231―236pp.

▶ THE MAGICAL CASTLE
迷路の城から脱出しよう。――菅原悟, マイコンBASIC Magazine, 7月号, 123―125pp.

MZ-700/1500

▶移植版 PAC-WORLD
ジャンプ台を利用して穴に落ちないようにゴールまで走れ!!――三井利靖, マイコンBASIC Magazine, 7月号, 121―122pp.

▶ 7ならべ
古典的カードゲーム「7ならべ」をパソコンで。――小笹籠一, マイコンBASIC Magazine, 7月号, 119―120pp.

MZ-80K/C/1200/700/1500

▶ MENTAL
なつかしのスペースインベーダーもどきのシューティングゲーム。――AO Jr., マイコンBASIC Magazine, 7月号, 116―118pp.

MZ-80B/2000/2200/2500/V2

MZ-2500

▶ FM音源エディタ
音色の変化がグラフ表示され, 作った音色をBASICで簡単に使える, FM音源エディタ。――小室真粧美, I/O, 7月号, 212―215pp.

▶新・パソコンサンデー活用研究
MZ-2500でキーの初期状態を変える。――高橋雄一, マイコン, 7月号, 239p.

▶なんでもQ&A シャープMZシリーズ編
アルゴエディタについて。――シャープ, マイコン, 7月号, 403―405pp.

▶移植版 奇跡の品物
チャオを操作し, 神経衰弱の要領でアイテムをそろえよう!!――恒田陽介, マイコンBASIC Magazine, 7月号, 131―133pp.

▶ザ・ゲーム・ミュージック・プログラム SILPHEED
シルフィードのBGMです。――3A18, マイコンBASIC Magazine, 7月号, 178―180pp.

▶ Computer Graphics Animation 第3回 サーフェイスモデリングツール
スムーズシェーディングまでサポートしたサーフェイスモデリングツール。――山下馨, ASCII, 7月号, 225―230・325―339pp.

▶ MZ-2500に2HD/2DDドライブを
MZ-2500に2HDドライブを接続するためのインタフェイスの製作。――ANOM/中山厚紀, I/O, 7月号, 140―144pp.

MZ-2500V2

▶天文計算プラネタリウム
今夜あるいは任意の夜の星空案内を音楽付きで楽しめるプラネタリウムプログラム。――植田庸一, マイコン, 7月号, 255―266pp.

MZ-80B/2000/2200/2500

▶タマゴツブシ
木の上のカラスのタマゴを下からつついて壊せ!!――井上智紀, マイコンBASIC Magazine, 7月号, 126―127pp.

MZ-2000/2200/2500

▶ Change Color
あるパターンに従って色が変化する妖精をつかまえよう。――小尾太志, マイコンBASIC Magazine, 7月号, 128―130pp.

MZ-2861

▶ MZ-2861上で動くエミュレータ・ソフト「MZEX」
98エミュレータ「MZEX」でメジャーなソフトが走るかどうかをチェック。――編集部, I/O, 7月号, 226―227pp.

▶ TEST ROOM MZ-2861 PC-9801エミュレータを検証
今回はMS-DOS, ワープロ, 98エミュレータなどソフトウェアについて解説する。――編集部, ASCII, 7月号, 191―194pp.

▶最新16ビットパソコンロードテスト
PC-286, MZ-2861, PC-9801VX2の3機種について, ベンチマークその他によって比較を試みる。――編集部, マイコン, 7月号, 154―158pp.

▶シャープ MZ-2861 エミュレーションモード
PC-98エミュレータの概要と使い方および実際に走るソフトを紹介。――髙橋雄一, マイコン, 7月号, 147―153pp.

X1/C/D/F/G/turbo/II/III/Z

X1turbo

▶高速ライン・ルーチン
HuBASICの5～8倍の速さで線を描画します。――U.K UOTA, I/O, 7月号, 206―211pp.

▶標準ディスプレイで高解像度表示を!
400ラインモードでスーパーインポーズを可能にします。――今雪寛, I/O, 7月号, 136―139pp.

▶なんでもQ&A X1/X1turbo/X68000シリーズ編
X1turboのシフトキーとCOPYキーを同時に押すとしばらく止まってしまうわけ。――シャープ, マイコン, 7月号, 401p.

▶簡易住所録SYS
葉書にあて名をプリントアウトできる住所録プログラム。――折原美昭, マイコン, 7月号, 348―356pp.

▶新・パソコンサンデー活用研究
X1turboでグラフィックを消さずにBASICを起動する。――高橋雄一, マイコン, 7月号, 239p.

▶新・パソコンサンデー活用研究
X1turboの2Dディスクを速くする。――高橋雄一, マイコン, 7月号, 238p.

▶なんでもQ&A X1/X1turbo/X68000シリーズ編
X1turbo/IIにCZ-520Fをつないだ場合に, 2HD, 2DDでBASICを起動する方法。――シャープ, マイコン, 7月号, 401―402pp.

X1シリーズ

▶ゲームソフト改造コーナー
信長の野望(全国版)で自国を強くする。――南紀白浜, Hacker, 7月号, 97p.

▶新・パソコンサンデー活用研究
X1で倍精度の表示桁数を増やす。――高橋雄一, マイコン, 7月号, 238―239pp.

▶もうすぐ夜明け
ファンタジックなアクションパズルゲームだ。――はりくんTM., マイコンBASIC Magazine, 7月号, 166―167pp.

▶ DESERT MAN
手榴弾を手にホリョを救い出せ!!――NAMKO 2, マイコンBASIC Magazine, 7月号, 164―165pp.

▶ザ・必勝法+㊙改造法 ディーヴァ
ゲストプレイヤーにクリシュナ・シャークがくるパスワードを公開!――安藤純, テクノポリス, 7月号, 125p.

▶なんでもQ&A X1/X1turbo/X68000シリーズ編
モデムターミナルCZ-133SFでBBSの内容をプリンタに出力する方法。――シャープ, マイコン, 7月号, 402p.

X68000

▶もうZ'sSTAFFなんて呼べない
驚異的な機能を持ちプロをも満足させるグラフィックツールZ'sSTAFF PRO 68Kなどを紹介。――編集部, LOGIN, 7月号, 204―207pp.

▶ X68000初のスーパーコミュニケーションツール, よしコムX
X68000用のターミナルプログラムだ。ソースリストで掲載。――編集部, LOGIN, 7月号, 240―243・334―338pp.

▶グラフィックで遊ぶX68000 BASIC
X68000のグラフィック機能とBASICからの使い方を解説。――編集部, POPCOM, 7月号, 110―115pp.

▶ X68000ハッカーズマニュアル
X68000のOSやアセンブラについての解説およびプリンタドライバのプログラム。――富田靖, I/O, 7月号, 257―263pp.

▶ X68000の全回路図
X68000の全回路図のほか, I/Oポートのアドレスや割り込み優先順位の表も掲載。――編集部, I/O, 7月号, 249―256pp.

▶ TEST ROOM X68000 プログラマのためのアセンブラの詳細②
先月に引き続きアセンブラの詳細および実際の開発例について。――編集部, ASCII, 7月号, 195―197・312―315pp.

▶ザ・ゲーム・ミュージック・プログラム WECル・マン24
ビデオゲームWECル・マン24のVGMです。――宮垣博守, マイコンBASIC Magazine, 7月号, 184―185pp.

▶なんでもQ&A X1/X1turbo/X68000シリーズ編
Human68kでのキー入力操作の効率アップの方法について。――シャープ, マイコン, 7月号, 400―401pp.

▶なんでもQ&A X1/X1turbo/X68000シリーズ編
システム起動後すぐにプログラムを実行する方法について。――シャープ, マイコン, 7月号, 400p.

▶ BASICプログラミングテクニック(裏ワザ・隠し機能編)
X-BASICのマニュアルにないテクニック, 拡張関数の作り方, そしてBASICを改造するサンプルを紹介。――高橋雄一, マイコン, 7月号, 180―186pp.

▶ X68000活用研究 ファンクションコール&IOCS活用法
IOCSの概要を公開し, ファンクションコールとあわせて使い方を解説。――塚越一雄, マイコン, 7月号, 172―179pp.

ポケコン

PC-1600K

▶ PC-1600K用逆アセンブラ
全バンク, 全アドレスを逆アセンブルできる, ザイログ表記の逆アセンブラ。――CASTER, I/O, 7月号, 145―147pp.

▶ポケコンを使いこなそう! 自動発信機能付電話帳プログラム
PC-1600Kに名前と電話番号を入力し, 検索したりモデムホンMZ-1X19で電話をかけたりできるプログラムだ。――塚田洋一, マイコン, 7月号, 338―343pp.

PC-1500

▶サンセット・フリップ
14種類の技を使い分けられる過激なプロレスゲーム。――あおきひばり, マイコンBASIC Magazine, 7月号, 171―172pp.

PC-1360K

▶一太郎の文章入力をポケコンPC-1360Kで!!
簡易ワープロプログラムで入力した文書を, レベルコンバータCE-130Tを用いて98のMS-DOSのファイルに転送。――和田岳雄, マイコン, 7月号, 359―361pp.

PC-1245

▶ BATTLER
鎧を着け, 盾と剣を持って敵と一騎討ち。――田中栄造, マイコンBASIC Magazine, 7月号, 170p.

編集室から

DRIVE ON

このコーナーでは本誌年間モニタの方々のご意見を紹介します。今月は6月号の記事に関する2期生最終回のレポートです。

●使いやすいOSとは，ユーザーが作ったプログラムを内部コマンドと同等の感覚で使えるものだと思います。ファイル操作はその要となる重要な存在。私はCP/MやMS-DOSなどを使ったことはありませんが，Human68kのファイル操作の手続きは，ユーザー側から見てかなり納得のいくものでしょう。それにしても，「Human 68k入門」くらいのことマニュアルに載せて当然だとも思いました。

深川　哲光（28）MZ-731/1500，X1G 香川県

●スペックを並べたてて「これが僕の理想のマシン」という投書をよく見かけます。ニューマシンに夢を託す気持ちはわかるし，私自身X68000がほしくてたまりません。しかしその前に，自分のマシンを使いこなす必要があります。代価さえ払えばどんなマシンでも手に入るけれど，マシンの性能にばかり頼っていてはパソコン文化は死んでしまう。ハードとともに我々も進化しなければならないとあらためて思いました。

渡辺　敦哉（20）X1F 埼玉県

ごめんなさいのコーナー

7月号　STORY　MASTER

カセットテープへのロードが正常に行われないという症状が発生しました。リスト1のルーチンを加えてください。これによりF（FS）のコマンドが追加されます。通常のデータにはFを，シークレットファイルにはFSをあらかじめ実行してロードを行ってください。また，乱数ルーチンに誤りがありました。以下の修正を加えてください。

3AB6H　57　→　4F
3AC7H　72　→　71

そのほか3008Hから3042Hまでのワークエリアにゴミが入っていました。気になる方は00Hで埋めてください。

7月号　試験に出るX1

「試験に出るX1」のMMLに以下のバグがありました。

1. 一部の音色が違う
2. 指定した音量によって音が出ない場合がある
3. 音長が間違っているのでテンポがずれてしまう

訂正方法は，リスト2を打ち込み“CHANGE”のファイル名でセーブしておきます。念のためにMMLプログラム全体のバックアップを取ったのちに，

NEW ON &HB800
LOADM “MML. OBJ”
LOAD “CHANGE”
RUN
SAVEM “MML. OBJ” &HA 8B0, &HAFFF

とします。

また，5月号125ページの図2-2でOP2とOP3のレジスタ番号が逆になっています。しかし上記の訂正により，「ソフト的には図2-2に従う」ようになりますので，Yコマンドを使う場合は図2-2に従ってください。

なおバグではありませんが，turboでない場合は6月号の53ページに従って変更してください。ここに再記しますと，

A965：A3　1F → 07　07
A9D6：A0　1F → 04　07
A9E5：A3　1F → 07　07

ということです。

6月号　Fuzzy BASICコンパイラ

3000H版のコンパイル後の動作に異常がありました。リスト3の修正を加えてください。

5月号　S-OS“SWORD”変身セット

TYPEコマンドで“More”のメッセージ出力後の動作に不備がありました。

33F6H　　CD　53　34

リスト1-1 STORY MASTER ダンプ部

```
350B 3A 54 35 CD A3 1F CD 09 : 28
3513 20 38 38 3E 3E CD F4 1F : EC
351B CD 9D 1F CD EE 1F 2A 08 : 95
3523 30 22 70 1F CD A6 1F 38 : AB
352B 22 2A 74 1F 11 32 30 01 : 53
3533 11 00 C5 E5 ED B0 E1 C1 : FA
353B 11 FF 31 ED B0 CD B9 33 : 97
3543 3A 54 35 FE 01 CC C1 35 : 84
354B C3 77 32 CD 33 20 C3 77 : C6
3553 32 00 ED 5B 76 1F 1A FE : 27
355B 46 C2 77 32 13 1A FE 53 : 2F
3563 20 04 3E 01 18 02 3E 04 : BF
356B 32 54 35 C3 77 32       : 27
---------------------------------
SUM: 62 59 A4 04 96 B9 AE 5E 63D8


3E69 AF 12 ED 5B 76 1F 3A 54 : 2C
3E71 35 CD A3 1F CD 09 20 DA : 94
3E79 4E 35 2A 06 30 22 70 1F : 94
3E81 E5 CD A6 1F DD E1 ED 5B : 7D
3E89 70 1F 2A 72 1F 19 22 08 : 8D
3E91 30 01 11 00 11 FF 31 2A : AD
3E99 74 1F ED B0 3A 54 35 FE : F1
3EA1 01 CC C1 35 C3 12 36    : CE
---------------------------------
SUM: 2C EC 49 F6 7D A9 75 D8 5607
```

32DØH　55　35

リスト1-2 STORY MASTER ソース部

```
0000                      1 %LOAD   EQU 350BH
0000                      2 %PASS2  EQU 3E69H
0000                      3 %GETCOM EQU 3612H
0000                      4 %ENDGET EQU 33B9H
0000                      5 %FLBUF1 EQU 3032H
0000                      6 %FLBUF2 EQU 31FFH
0000                      7 %HOT    EQU 3277H
0000                      8 %PFILE  EQU 35C1H
0000                      9 %TXSTAT EQU 3006H
0000                     10 %TXEND  EQU 3008H
0000                     11 #FILE   EQU 1FA3H
0000                     12 #ROPEN  EQU 2009H
0000                     13 #RDD    EQU 1FA6H
0000                     14 #ERROR  EQU 2033H
0000                     15 #FPRNT  EQU 1F9DH
0000                     16 #PRINT  EQU 1FF4H
0000                     17 #LTNL   EQU 1FEEH
0000                     18 #KBFAD  EQU 1F76H
0000                     19 #IBFAD  EQU 1F74H
0000                     20 #DTADR  EQU 1F70H
0000                     21 #SIZE   EQU 1F72H
0000                     22
0000                     23      OFFSET $850B-%LOAD
0000                     24 ;------------------
32D0                     25      ORG 32D0H
32D0 55 35               26      DW %FCMD
32D2                     27 ;------------------
350B                     28      ORG %LOAD
350B                     29 ;
350B 3A 54 35            30      LD A,(%FILE)
350E CD A3 1F            31      CALL #FILE
3511 CD 09 20            32      CALL #ROPEN
3514 38 38               33      JR C,%RDERR
3516 3E 3E               34      LD A,'>'
3518 CD F4 1F            35      CALL #PRINT
351B CD 9D 1F            36      CALL #FPRNT
351E CD EE 1F            37      CALL #LTNL
3521 2A 08 30            38      LD HL,(%TXEND)
3524 22 70 1F            39      LD (#DTADR),HL
3527 CD A6 1F            40      CALL #RDD
352A 38 22               41      JR C,%RDERR
352C                     42 ;
352C 2A 74 1F            43      LD HL,(#IBFAD)
352F 11 32 30            44      LD DE,%FLBUF1
3532 01 11 00            45      LD BC,17
3535 C5                  46      PUSH BC
3536 E5                  47      PUSH HL
3537 ED B0               48      LDIR
3539 E1                  49      POP HL
353A C1                  50      POP BC
353B 11 FF 31            51      LD DE,%FLBUF2
353E ED B0               52      LDIR
3540 CD B9 33            53      CALL %ENDGET
3543 3A 54 35            54      LD A,(%FILE)
3546 FE 01               55      CP 01H
3548 CC C1 35            56      CALL Z,%PFILE
354B C3 77 32            57      JP %HOT
354E                     58 ;
354E                     59 %RDERR
354E CD 33 20            60      CALL #ERROR
3551 C3 77 32            61      JP %HOT
3554 00                  62 %FILE DS 1
3555                     63 ;
3555                     64 %FCMD
3555 ED 5B 76 1F         65      LD DE,(#KBFAD)
3559 1A                  66      LD A,(DE)
355A FE 46 C2 77 32      67      IF A<>'F' JP %HOT
355F 13                  68      INC DE
3560 1A                  69      LD A,(DE)
3561 FE 53 20 04 3E 01 18 70     IF A='S' THEN LD A,1 ELSE LD A,4
3568 02 3E 04
356B 32 54 35            71      LD (%FILE),A
356E C3 77 32            72      JP %HOT
3571                     73 ;------------------
3E69                     74      ORG %PASS2
3E69                     75 ;
3E69 AF                  76      XOR A
3E6A 12                  77      LD (DE),A
3E6B ED 5B 76 1F         78      LD DE,(#KBFAD)
3E6F 3A 54 35            79      LD A,(%FILE)
3E72 CD A3 1F            80      CALL #FILE
3E75 CD 09 20            81      CALL #ROPEN
3E78 DA 4E 35            82      JP C,%RDERR
3E7B 2A 06 30            83      LD HL,(%TXSTAT)
3E7E 22 70 1F            84      LD (#DTADR),HL
3E81 E5                  85      PUSH HL
3E82 CD A6 1F            86      CALL #RDD
3E85 DD E1               87      POP IX
3E87                     88 ;
3E87 ED 5B 70 1F         89      LD DE,(#DTADR)
3E8B 2A 72 1F            90      LD HL,(#SIZE)
3E8E 19                  91      ADD HL,DE
3E8F 22 08 30            92      LD (%TXEND),HL
3E92                     93 ;
3E92 01 11 00            94      LD BC,17
3E95 11 FF 31            95      LD DE,%FLBUF2
3E98 2A 74 1F            96      LD HL,(#IBFAD)
3E9B ED B0               97      LDIR
3E9D 3A 54 35            98      LD A,(%FILE)
3EA0 FE 01               99      CP 01H
3EA2 CC C1 35           100      CALL Z,%PFILE
3EA5 C3 12 36           101      JP %GETCOM
3EA8                    102 ;
3EA8                    103 ;----------------------------------
3EA8                    104 ;ｱﾀﾗｼｸ ﾂｲｶ ｻﾚﾀ ｺﾏﾝﾄﾞ(ｴﾃﾞｨﾀ)
3EA8                    105 ; F    ｱｽｷｰﾌｧｲﾙ ｦ ﾖﾑ ｼﾞｭﾝﾋﾞ ｦ ｽﾙ.
3EA8                    106 ; FS   ｼｰｸﾚｯﾄﾌｧｲﾙ ｦ ﾖﾑ ｼﾞｭﾝﾋﾞ ｦ ｽﾙ.
3EA8                    107 ;----------------------------------
```

3453H　　F5　CD　F4　1F　F1　C9

に変更してください。またMZ-2500でRUNコマンドの際のG-RAM退避アドレスが誤っていました。

FE96H　　3C　→　3D

に変更してください。

リスト2　MML

```
100 MEM$(&HA8CB,5)=HEXCHR$("C9 F5 FE 48 38")
110 MEM$(&HA8D0,8)=HEXCHR$("0E 47 87 E6 30 EA DE A8")
120 MEM$(&HA8D8,8)=HEXCHR$("D6 18 ED 44 80 47 78 01")
130 MEM$(&HA8E0,7)=HEXCHR$("00 07 CD C7 AD F1 C9")
140 MEM$(&HAC99,3)=HEXCHR$("C3 E9 A8")
150 MEM$(&HAC9D,3)=HEXCHR$("C3 F0 AF")
160 MEM$(&HAD68,3)=HEXCHR$("C3 D9 AF")
170 MEM$(&HADC4,3)=HEXCHR$("C3 CC A8")
180 MEM$(&HAFA6,1)=HEXCHR$("00")
190 '
200 FOR A=&HAE23 TO &HAE63 STEP 2
210   D=CVI(MEM$(A,2))-1
220   MEM$(A,2)=MKI$(D)
230 NEXT
```

リスト3　コンパイラ

```
3000 C3 F1 3F
3FF1 ED 73 FD 3F ED 7B 6A 1F
3FF9 CD 00 40 31 00 00 C9
```

バグに関するお問い合わせは
☎03(263)2230(直通)
月～金曜日16：00～18：00

お問い合わせは原則として，本誌のバグ情報のみに限らせていただきます。入力法，操作法などはマニュアルをよくお読みください。

また，よくアドベンチャーゲームの解答を求めるお電話をいただきますが，本誌ではいっさいお答えできません。ご了承ください。

地球は暑かった 日本語ワープロで 熱くなろう

▼およそキーボードにふさわしくない，と，時に思えてしまう日本語ですが，今月の特集ではこの問題をテーマにしました。

メーカー各社には心して読んでもらいたいワードプロセッサの評価や，各スタッフによるエディタを含めた料理法など，いろいろな方面から「迷宮の日本語処理環境」に切り込みを入れています。最後には，市販ソフトも顔負けしそうなMZ-2500用のプログラムも登場しました。読者の皆さんの感想はいかがでしょうか。

かなや漢字の混在する日本語の処理には，変換作業をしなければならないことから，欧文ワープロよりはるかに多くのことを要求されます。現在の日本語処理環境は，アルファベット言語のそれに比べると確かに逆境といえるかもしれません。ですが，「逆境」の中で磨かれてこそ，なにごとも光を放つようになるもの。ワープロをはじめとするこの処理環境が，今後もどのように進化していくか，おおいに期待しましょう。

▼発表されてから2年以上のたったS-OSが，またその輪を広げました。FM-7/77版S-OS“SWORD”。パズルゲームの碁石拾いやグラフィックツールを使う漢字出力パッケージJACKWRITEも加わっています。また，「ユーザーが育てていくマシン」を体現しているX68000には，新たに「X68000BASIC入門」が始まりました。ご意見をお待ちしています。

▶関東地方をはじめ，出足の遅かった今年の梅雨は，7月に入ってからも各地で深刻な水不足を招きました。「この週末には天気も下り坂になり，待望の雨となるでしょう」とはあるときの天気予報の弁。たとえ恵みの雨であっても，やはり「下り坂」と表現されてしまうのではなんだか雨雲に気の毒な気がして，思わず苦笑いしてしまいました。

海の向こうからは，太陽系第10惑星発見か，のニュースも入っています。白鳥座がメインステージに座すこの季節，たまには夜空をじっくりながめるのもいいでしょうね。

▶本誌で好評のエッセイ「知能機械概論」は，筆者の都合により，残念ながら今回は休載となりました。来月号からは復帰しますのでまた応援をお願いします。

投稿応募要領

●原稿には，住所・氏名・年齢・職業・連絡先電話番号・機種・使用言語・必要な周辺機器・マイコン歴を明記してください。

●プログラムを投稿される方は，詳しい内容の説明，利用法，できればフローチャート，変数表，メモリマップ（マシン語の場合）に，参考文献を明記し，プログラムをセーブしたテープ（ディスケット）を添えてお送りください。また，プログラムは最低2回はセーブしてください。

●ハードの製作などを投稿される方は，詳しい内容の説明のほかに回路図，部品表，できれば実体配線図も添えてください。編集室で検討の上，製作したハードが必要な場合はご連絡いたします。

●投稿者のモラルとして，他誌との二重投稿，他機種用プログラムを単に移植したものは固くお断りいたします。

あて先

〒102 東京都千代田区九段南2-3-26井関ビル
日本ソフトバンク出版部
Oh! MZ「㋢㋮㋳」係

SHIFT BREAK

▶聞くとほっとする音楽というのがある。僕の場合はザ・バンドがそれだ。バンドの音楽ってのは泥臭いし古めかしい。が，妙に暖かい。なんていうか，アメリカの広さを感じさせる。僕の世代かそれ以上の人で，レコード棚の隅っこにバンドのアルバムが埋もれていたら，今夜あたりほこりを払ってそっと針を落としてみたらいかがだろう。（T.T.）

▶某月某日……カーマインX1，大戦略II，グラディウスとゲームのはしごをしたら目はチカチカ頭はボケボケ顔はゾンビの三重苦。某月某日……ビートロックのライブとクラシックのコンサートをはしごしたら，あまりの客層の違いに音楽とはなんぞやと悩むことしきり。皆さん，健康と美容のために1日の予定はひとつにしましょう。（K.Y.）

▶毎日のように水不足を伝えるニュースが流れている。「○○ダムの水が枯れるまであと××日」などとまさに『宇宙戦艦ヤマト』を思い出させるような悲壮感に満ちたレポートもある。しかし私には帰る田舎がある。たとえ水道の水が出なくなっても井戸水だけで生きていけるところなのである。というわけでさっさと東京を脱出することにしよう。(MOK)

▶X68000の使い方。FINDとSORTを使って簡易辞書を作りましょう。論文なんてものはその中に使ってある単語がだいたい決まっています。あれっ？　この単語なんて意味だっけと思ったときにあの厚い英和辞典をいちいち引くのはとっても面倒。これなら辞書を引くという一番面倒なところをコンピュータが受け持ってくれる。う～ん，楽ちん楽ちん。(IMT)

▶オー，Pascal，あなたのお家はどこ。わたしのお家はスイッツランドよ。ALGOLとModulaのとなりなのよ。ヤッホー　FORTRAN，RAN，RAN，ヤッホーFORTRAN，RAN，RAN，ヤッホ，FORTRAN，RAN，RAN，ヤッホホ。というわけで，turbo-Cの上陸が近い。ラムはするめが好きだった。（G）

▶忘れやすい名曲シリーズ
（知らない人は困ったチャンですよお）
「オ～，ジャンボリージャンボリジャンボリージャンボリジャンボリージャンボリアハハハハハハ，ジャンボリージャンボリジャンボリージャンボリジャンボリージャンボリアッハッ！」
うーんボーイスカウト（K.S.）

▶「へっ，ふ・う・し・ん？」いくら発熱がして全身に発疹ができたからといって，そりゃあんまりだ。というわけで，医者から1週間の外出禁止を言い渡された僕は，いわゆる自宅療養中の身。これで日頃の睡眠不足が解消できるぞと思ったものの，夏休みよりも長い休暇（？）に暇をもてあます今日この頃です。ああ，もう有休がないよお。（隔離のKO）

▶ご存じの方がいたら教えていただきたいのですが，利害関係のない第三者による「ＴＲＯＮの学術的評価」というものは存在するのでしょうか？　もしあったとしたら，どのような内容でどこに発表されたものなのでしょうか？　日本のマスコミは頻繁にＴＲＯＮを取り上げますが，本当にそれだけの価値があるものなのですか？　誰か教えてください。（M）

▶俳優のクリストファー・ウォーケンは，黙っているとただの根暗なおじさんだが，笑うととても魅力的になる。だから好きなんだと話したら，Ｋ．Ｙ．氏に「なんで女ってウォーケンファン多いの？」といきなりかみつかれた。どうやら彼のガールフレンドが私と似たようなことをいったらしい。そりゃあ面白くないだろうな。納得。（よ）

▶つい最近X68000を買ったばかりなのだが，先日MZ-2500V2を買ってしまった。ついでにモデムも買ってパソ通などにも手を出している今日このごろ。んで，Oh!FMネットで初めてチャットした相手というのがOh!FMの編集長だった。うーん，ぜんぜん新鮮味がない。結局，まだ電話以上の使い方は見つけていないのでした。（U）

▶先日，夜の横浜の街をうろうろしたあと，山下公園が見える喫茶店に入ってぼんやりと点滅する燈台の灯りを眺めていました。そこまではよかったのですが，なにしろ今月はワープロ特集だったせいもあって朝から晩までずっとワープロを打っていたものだから，点滅する灯りをじっと見ていると，それが突然ディスプレイのカーソルの点滅へと豹変して見えてしまったのです。ああ，怖かった。（N）

▶コマンドモードで立ち上げてAUTOEXEC.BATで辞書をRAMディスクに転送してからビジュアルシェルを起動する。いつもならここでワープロに行くのだが，きょうはBASICでプログラミング。行き詰まったらチャイルドプロセスでグラディウスをすればいい。BASICに帰ればちゃんとテキストは残ってる。早い人はもう体験してる，2Mバイトの世界。（@）

▶その昔，ラインエディタよりも使いにくいポータブルワープロがどうしてそんなに売れるのかよく質問されたものだ。「だってキレイな漢字が印刷できれば自慢できるじゃない」と答えながらも私は，そのまた昔のダイモのブームを思い出していた。ところで梅雨はどこにいったのだろう。「雨の日はふたりで」なんてせっかくアンアンがレイングッズの特集を組んでくれたのに……。（T）

microOdyssey

毎年，夏になると反戦映画が公開される。特に核戦争を扱ったものが多く，この夏もイギリスのアニメ映画「風が吹くとき」という作品が話題になりそうだ。この作品では老夫婦の生活をとおして，おもに放射能の恐怖というオーソドックスな視点で核の脅威を訴えている。実際これまでは核戦争＝放射能というのが長年の図式だった。ところが最近は放射能以上に「核の冬」というものを心配している人も多いようだ。

この核の冬という概念が登場したときは核戦争の新たな脅威として世論を盛りあげ，核の抑止につながると期待する向きもあった。核の冬を例にあげ核戦争に勝者なし，核の廃絶をと訴える声と，核の冬は回避せねばならないという声が同列で叫ばれていたのだ。そして，この核の冬という現象を抑止するための研究は大いになされることになった。しかし，それは核戦争を抑止することとはなんの関連もない研究だったといえよう。

そもそも核戦力の増強は核に対する防備と表裏一体の関係にある。たとえば命中精度の高いミサイルが作られれば，それと同等の兵器による攻撃に耐えるシェルターが作られるのが当然だ。すなわち核戦争の当事者は常に安全が保証されていなければならない，それが核競争のルールだ。しかし，核の冬という概念の登場により，核の第1撃に耐え放射能の脅威が薄れたあとの安全が保証されなくなってしまった。その結果の騒ぎが2年前の「核の冬」だったのだ。当然，その対策も核戦争の当事者の立場に立ったものでしかない。膨大なシミュレーションによる核の冬の進行分析や状況分析は水爆の直撃にも耐えられるごく一部の人々のためであって，残留放射能を防ぐすべもない多くの人々のことは初めから考慮されていない。世界のほとんどの人にとって核の冬など，太陽の超新星化以上の心配は必要ないものだったのだ。

さらに危険なことは，これらの研究は「いかに安全に核戦争を行うか」という視点にのみ基づいていることだ。つまり，核の冬を起こさずに核戦争を行うためにはどうすればよいのか，爆発の粉塵を高空まで吹きあげないようにするにはどの程度の核爆弾を使用すればよいのか。そこには核兵器を使用しないようにするという視点は最初から欠如している。あくまでも核戦争をするならという前提に立っているのだ。その成果として「正しい核戦争の行い方」というものも明らかになってきていることだろう。

そして今日，かつてよりははるかに核戦争というものが現実的なものとなった。以前はまったく未知のものとして空想するしかなかった核戦争後の世界というものが，かなり正確であろうといわれるレベルまでシミュレートされるようになったからだ。このことが核戦争の抑止として働くならば，十分に意味のある論議だったとして評価できるだろうが，逆に全面核戦争はともかく小規模の核戦争なら手軽にできるといった一種の安心感を与えてしまったようにも思われる。これではかえって核戦争がやりやすくなってしまっただけではないのか。結局，2年前から真剣に行われていた論議はいったいなんだったのだろうか。核の脅威というものが取りあげられるたびに疑問が頭を持ちあげてくるのだ。

(U)

1987年9月号8月18日(火)発売
特集　MZ-700に不可能はない
MZ-2500 MMLのグレードアップ
PC-8001/8801版S-OS“SWORD”発表

バックナンバー常備店

地域	市区	書店	電話
東京	神保町	三省堂神田本店5F	03(233)3312
	〃	書泉ブックマートB1	03(294)0011
	〃	書泉グランデ5F	03(295)0011
	八重洲	八重洲ブックセンター3F	03(281)1811
	新宿	紀伊国屋書店本店	03(354)0131
	高田馬場	未来堂書店	03(200)9185
	渋谷	大盛堂書店	03(463)0511
	池袋	西武百貨店11Fブックセンター	03(981)0111
	〃	西武百貨店9F コンピュータ・フォーラム	03(981)0111
	町田	久美堂東急ハンズ店	0427(28)2783
神奈川	横浜	有隣堂横浜駅西口店	045(314)9726
	〃	有隣堂ルミネ店	045(453)0811
	〃	横浜書店	045(241)5445
神奈川	藤沢	有隣堂藤沢店	0466(26)1411
	厚木	有隣堂厚木店	0462(23)4111
	平塚	文教堂四の宮店	0463(54)2880
千葉	柏	新星堂カルチェ5	0471(64)8551
	船橋	西武百貨店10Fブックセンター	0474(25)0111
	〃	東京旭屋書店船橋店	0474(24)7331
	〃	芳林堂書店津田沼店	0474(78)3737
	千葉	多田屋千葉セントラルプラザ店	0472(24)1333
	木更津	松田屋	0438(23)4210
大阪	都島区	駸々堂京橋店	06(353)2413
埼玉	川越	黒田書店	0492(25)3138
	川口	岩渕書店	0482(52)2190
茨城	水戸	川又書店駅前店	0292(31)0102
京都	中京区	オーム社書店	075(221)0280
長野	飯田	平安堂飯田店	0265(24)4545
北海道	室蘭	室蘭工業大学生協	0143(44)6060

定期購読のお知らせ

定期購読の申し込みをお受けしています。本誌が手に入りにくい地区にお住まいの方，毎月購読していただいている方，入手確実な定期購読への加入をお勧めします。詳しくは，本誌とじ込みの振替用紙をご覧ください。

バックナンバー在庫状況
1986年10，11，12，1987年1，2，3，4，5，6，7以上の在庫がございます。

バックナンバーのご注文はお近くの書店からできますが，どうしても入手しにくい場合，直接弊社へ現金書留にてご注文ください。なお，郵送料は冊数によって異なりますので，前もってご連絡ください。お問い合わせは，出版営業（☎03-261-4095）宛お願いします。

海外送付ご希望の方へ

本誌の海外発送代理店，日本IPS㈱にお申し込みください。なお，購読料金は郵送方法，地域によって異なりますので，下記宛必ずお問い合わせください。

日本IPS株式会社
〒101 東京都千代田区神田小川町3-5
☎03(291)2632

Oh!MZ 8月号

■1987年8月1日発行　定価480円　■発行人　孫　正義　■編集人　笹口幸男
■発売元　（株）日本ソフトバンク
■出版事業部　〒102 東京都千代田区九段南2-3-26 井関ビル　☎03(261)4095　FAX 03(262)8397
編集室☎03(239)4156
出版営業☎03(261)4095
広告営業☎03(255)9677
■本　社　〒102 東京都千代田区九段南2-3-14　靖国九段南ビル　☎03(263)3690㈹
TELEX 東京　232-4614JSBTYJ　FAX 03(263)3660
■西日本営業部　〒541 大阪府大阪市東区南本町2-6　明治生命堺筋本町ビル10F
☎06(264)1471㈹　FAX 06(264)1481
■印　刷　凸版印刷株式会社

ところで君のマークⅢは元気かな？
セガ・マークⅢの疲労回復に
もっとも効果があるといわれている
『セガ・ハイテク図鑑』が
いよいよ発売されます。
「BEEP」パワー総結集！
NEW RELEASE
THE SEGA HIGH TECHNIC PICTURE BOOK
『セガ・ハイテク図鑑』は
次の症状の人に効果があります。
☞"ぴかぴか"の新作「ザ・プロ野球 ペナントレース」や「アウトラン」をもっと楽しみたい人
☞難解「ロレッタの肖像」や「スケバン刑事Ⅱ」を途中であきらめてしまった人
☞「スペースハリアー」の完全攻略と裏ワザのすべてを試してみたい人
☞スポーツゲーム「ロッキー」と「グレートバレーボール」でひと汗かきたいと思っている人
☞「赤い光弾ジリオン」「魔界列伝」「スーパーワンダーボーイ」でどうしても先に進めない人
新鮮なビタミン技と攻略エキス配合の
『セガ・ハイテク図鑑』は、
B6判/112ページで定価600円。
7月17日(金)発売予定です。
SOFT BANK
日本ソフトバンク出版事業部
〒102 東京都千代田区九段南2-3-26井関ビル TEL03(261)4095

X1/X1turboシリーズテクニカルノウハウ

X1-Techknow

[エックスワン・テクノウ] **B5判** 定価3,900円

BNN第二企画部編

新刊

大好評のTechknowシリーズ第3弾は、ホームパソコンとして発売以来絶大な人気を誇る〈X1シリーズ〉。〈ターボシリーズ〉へと続く一連のラインナップは、数多くのX1ソフト資産を継承しつつ、様々なユーザーを魅了してきました。本書はX1の持つポテンシャルを最大限に活用し、プログラム作りの楽しさを肌で感じるためのテクニカルノウハウ書です。

アーキテクチャから周辺デバイス、ディスク制御、画面制御、RS-232Cを始めとする各種インターフェイスの活用法など豊富な図表とサンプルプログラムと共に詳しく解説します。

目　次

商品の詳しい内容をお知りになりたい方は、資料請求券を添付し書名明記の上、下記の宛先まで葉書でお申込み下さい。折り返し資料をお送り致します。

株式会社ビー・エヌ・エヌ　東京都千代田区麹町4-5　紀尾井町レジデンス5F　お問い合せ03(238)1321営業部まで

新・発・売・!!

定価 ¥29,000

財計くん 売掛管理台帳

出力帳票：売上日計表・残高一覧表・請求書・領収証
顧客コードNo,一覧表・取扱商品一覧表・DMシール

管理顧客数	1データーディスク内に600名までです。
取扱商品数	〃 に150品目までです。
顧客1人当り売上件数	月/60件までです。ラクラク金額入力でカンマ付、無のどちらもOK。
請求書・領収証	貴社の当月分の「お知らせ」が出力できます。
DMシール	条件検索して発行できます。
〆切繰越	一括方式と個別方式との両方が可能です。
プリンター用紙	白紙11インチ、又はヒサゴGB342を使用願います。

この台帳は貴方の実務サイズです。各機能は貴社の実務を軽々サポートします。

X1 turbo OK-システム 漢字

X-1ターボ 専用

発・売・中・!!

定価 ¥32,000

DATA・CARD・1200

1. カード型データーベースとして

検　　索：1124枚のデーターカード内から3重条件を処理します。
データー入力：自由設定項目12個をフルに活用、各データーは漢字（全角文字）で最長20字まで使用可能です。勿論、追加・変更・削除もOKです。当然データーディスクの作成は思いのままです。
表示&印刷：検索機能は当然。その上カード・ソーティング機能との連係での、DMシール、業者宛名、カードNoによるデーターの抜粋、ステップ印刷が可能です。

2. グラフ・カードを活用した、グラフ・データーファイルとして

表示&印刷：7種類・22タイプのグラフを作成します。12項目12データーを1単位として1枚のグラフ・データーディスクに76個を格納し、処理します。

※縦棒グラフ・横棒グラフ・帯グラフ・円グラフ・折線グラフ，各棒グラフは3D仕様可能です。

62年4月より、データー変換ツールを内蔵した、バージョンアップ版で出荷しております。なおバージョンアップ版への変換は2,400円を申し受けます。

定価 ¥39,800

個人簿記会計 財計くん

出力帳票　仕訳帳・期首試算表・期末試算表・貸借対照表
損益計算書・各科目別元帳・科目コード一覧表
摘要コード一覧表・合計残高試算表

データーは高速処理	各帳票は約45秒で作成します。
仕訳入力は一度	振替伝票による一括入力方式を採用しています。
オート・ソース	仕訳訂正を実行すれば日付順でデーターを並べ変えます。
ラクラク金額入力	カンマ付、無のどちらでも受付ます。

金額処理は9桁10億円まで。仕訳件数は月／900件。
勘定科目はすべてコード入力で75個まで使用できます。
摘要小書きコード入力の〔A〕と自由入力〔B〕との二つで処理を対応しています。

財計くんは導入されたその日から貴社のオリジナルソフトに変身します。

OKハウス
〒885 宮崎県都城市都島町430-2

●関東受注センター TEL. 03(226)7234
●関西受注センター TEL. 06(375)3197
●開発センター TEL.0986(25)0303
各受注センターは24時間受付。開発センターは日曜はお休みします。

開発センター
〒885 宮崎県都城市都島町430-2
振込口座　鹿児島銀行都城支店
普 396174　大木芳幸

※各資料の御請求は、200円切手を同封して開発センターへお申し込み下さい。デモサンプルはそれぞれ2,400円を申し受けます。

自作派のあなた!!
パソコン通信はＢＢＳだけではありません。

SUPER DEVICE MONITOR "T" の実行例

いま流行のパソコン通信はカタカナだけか，あるいは漢字の混じった文章と簡単なグラフィクスだけだと思っていませんか。新発売の『SUPER-DEVICE-MONITOR"T"』を使えば，パソコン通信で機械語のソフトや，グラフィクスのバイナリィ・データを，特殊なデータ圧縮法により，セクター単位に最高通常の３２倍（理論値）の高速でアクセスが出来ます。これから発売予定の他機種用の『SUPER-DEVICE-MONITOR』シリーズとの互換性を考えて，Super MZ が使える総てのボーレートに対応し，ディバイス・エディターとしての機能や操作性なども各種ディバイスのデータを，瞬間的にセクター単位に表示，書き替え，検索，転送などが出来る事で，今まで大好評発売していた『スーパー修理屋さん』の最上位バージョンですので安心してお使い戴けます。

新発売

SUPER DEVICE MONITOR "T"

MZ-2500 全シリーズ 3.5″

13,000円

ゲーム派のあなた!!
知っていますか？便利なソフトの整理箱

アナタはテープ版のソフトを何本持っていますか？ソフトの中にはテープ版しかない物も少なくありませんが，テープ版はロード中，長い時間イライラ待たされたり，１本に１つのソフトしか入っていないので，何本もテープを持っていると，どんなに整理してあっても，使いたいソフトを見付けるだけで，時間をムダにする事も度々です。そんなアナタのために，市販のディスク一枚の中に，最高１７本のＩＰＬのテープ版ソフトを収容出来，多分割ロードのソフトでも，まるでディスク版ソフトの様に，スイッチＯＮからプレイ開始まで数秒で起動出来る『EXTRA-HYPER＋α』がお役に立ちます。扱えるソフトのタイトル数はＸ１の場合は１５２種，ＭＺは２６種類以上です。『EXTRA-HYPER＋α』があればテープ版ソフトの整理が出来て，イライラ解消の一挙両得です。

EXTRA HYPER＋αの実行例
画面中のソフトは同梱ではありません。

EXTRA-HYPER＋α

X1turbo・X1シリーズ 5″・3″
MZ-2000／2200 5″
MZ-2500（2000モード） 3.5″

X1（マニア・タイプ）・MZ-2000は要G-RAM **各14,000円**

お求めは全国の有名マイコンショップでどうぞ。
通信販売をご希望の方は当社へ直接、商品名・機種名・メディア名住所・氏名・電話番号を明記の上、現金書留にてお申し込みください。（送料無料）

BLUE SKY Co.

株式会社 BLUE SKY
〒411 静岡県三島市加茂16-4
☎0559-72-6710

夏！ツクモ Vol.2 ザ・バーゲン

半期に一度の
決算セール！
7/25〜31

安心のツクモシステム

おかげさまで創業40年 Thanks 40th TSUKUMO

代金引き換え配達
商品到着の際、お宅の玄関でお会計ができます。

ツクモらくらくクレジット
月々¥3,000以上の均等払いも頭金なし。ボーナス月加算払いを併用して欲しいもの先取り！
☎03-251-9911へ‼

秋葉原電気まつり

いよいよ8月2日まで
1等10万円！
5000円以上お求めの方に抽選券進呈

パーソナルワークステーション X68000

Good Selection TSUKUMO

CZ-600CE	本体＋キーボード	¥369,000
CZ-600DE	15型カラーディスプレイ	¥129,800
CU-15M1E	15型カラーディスプレイ	¥99,800
CZ-6VT1	カラーイメージユニット	¥69,800
CZ-6PV1	カラービデオプリンタ	¥198,000
CZ-6BE1	1MB増設RAMボード	¥35,000
CZ-6EB1	拡張I/Oボックス	近日発売
CZ-6BG1	GP-IBボード	近日発売
CZ-6BU1	ユニバーサルI/Oボード	近日発売

シャープ「EXE」ショップのツクモがシャープファンにおくる
X68000 ユーザーのための講習会

- ツクモサービスセンター3F
- 8月2日(日) 午後1時〜
- 参加費用 1,000円(お飲物、会員優待券、テレホンカード他付)
- 定員30名(先着)
- 協力 サムシンググッド他

講習会のお申し込みは
7号店✆03-253-4199(荒井)へ

X1F モデル10セット

- CZ-811C …… ¥89,800
- TX-12T1 …… ¥67,800
  (ナショナル12インチモニター)
- オリジナルソフト(3本) …… サービス

合計定価¥157,600
特価¥44,800

初回¥7,988	月々¥7,900×5回
冬のボーナス一括払いできます	

「ツクモおすすめモデム」

田村電機(製造元)
VERSA MODEM1200
定価¥59,800
ATコンパチ300/1200全二重
RS-232Cケーブル
サービス
特価¥20,000

ツクモオリジナルマウス TS-MX1

(X1ターボシリーズ／MZ-2500用)
特価¥6,800
X1・X1F・X1GにはCZ-8BM2(¥19,800)が必要です

ツクモオリジナル TS-FD MKII 5インチ2D

送料¥1,000

【1ドライブ】 定価¥44,800 **特価¥31,800**
【2ドライブ】 定価¥66,800 **特価¥49,800**

- FM77D2/L2/L4/AV用として……ケーブル(FM-77CA別売¥9,800)でドライブ番号切替可能、今までの5インチソフトもそのまま使えます。(FM-7/NEW7のシングルドライブとしては使用できません。)
- FM11AD2⁺・MB27607用としても使用できます。お問い合せ下さい。
- X1シリーズ、MZ-2500用として…別売ケーブル(TS-MXCA ¥5,000)が必要です。お申し込みの際はドライブ番号をご指定下さい。

通信ソフト SPS JETターボターミナル

150〜9600ボー対応、オートダイヤル、オートログイン、アップロード、ダウンロード、パラメータ設定、VT-100エスケープシーケンス対応
¥9,800

X1G モデル10セット

- CZ-820C …… ¥69,800
- CZ-811D …… ¥89,800

合計定価¥159,600
特価¥79,800

X1 turbo Z セット

- CZ-880C …… ¥218,000
- 15インチマルチスキャンテレビ ¥128,000

合計定価¥346,000
特価¥275,000

初回¥11,750	月々¥10,000×23回 ボーナス加算¥20,000×4回
冬のボーナス一括払いもできます	

CZ-8PD3
9ピンドットプリンター 定価¥59,800
特価¥29,800

CZ-8PR2R
カラープロッタプリンター 定価¥59,800
特価¥9,800

CZ-8PK2
16ドット漢字プリンター 定価¥134,800
特価¥39,800

CZ-8DT
デジタルテロッパー 定価¥89,800
特価¥16,800

X1シリーズ周辺機器 送料別途

品	名	定価	特価
MZ-1P17	カラー熱転写漢字プリンターケーブル付)	¥79,800	¥42,800
CZ-8BV2	カラーイメージボード2	¥39,800	¥33,800
CZ-8RL1	データレコーダ	¥24,800	¥21,000
CZ-8BE2	320KB外部メモリ	¥29,800	¥25,300
CZ-8BK3	第2水準漢字ROM&ソフト	¥13,800	¥11,700
CZ-8BK4	X1turboII用第2水準漢字ROM	¥ 6,800	¥ 5,800
CZ-8EB3	拡張I/Oボックス	¥33,800	¥28,700
CZ-8BR1	立体映像セット	¥29,800	¥25,300
CZ-8BM2	マウス&RS232Cボード	¥19,800	¥16,800
CZ-8BGR2	CZ-850C用G-RAM	¥14,800	¥ 4,800

MZ-2500用周辺機器 送料別途

型番	品名	定価	特価
TS-V25	ビデオ-RAM(64KB)	——	¥ 9,000
TS-M25	増設RAM(128KB)	——	¥ 8,200
TS-VM25	増設RAM＆ビデオ-RAM	——	¥16,800
MZ-1R28	辞書ROMボード	¥13,000	¥12,800
MZ-1R37	RAMディスクボード(640KB)	¥34,800	¥29,600

下取り・中古・もちろん新品もツクモニューセンター店へ

☎03-251-0987
〒101 東京都千代田区外神田1-16-10

▲ツクモニューセンター店は

1. **下取りグレードアップができます。**
   店頭へ持ち込み又は運送便でお送り下さい。差額査定後連絡いたします。
2. **中古品を扱っています。**
   希望中古品の予約もできます。
3. **トレードシステム(完全委託販売)**
   お客様に代って希望価格で販売します。下取りはもったいないという方にはピッタリです。
4. **ツクモオリジナル商品は全て展示。**
   ハード&ソフト何でもご相談下さい。
5. **新品・中古全国通販いたします。**
   ニューセンター店に☎を！
6. **24時間中古情報ダイヤル☎03-251-9977**
   いつも新鮮なニュースがいっぱい！

中古品をご希望の方は☎で在庫を確認して下さい。

限定！20セット X1 turbo (中古)

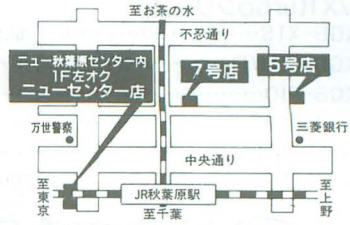
- モデル30(CZ-852CE) ¥70,000
- 専用モニター(CZ-850D) ¥50,000

ムダのない かしこいお買い換え 下取り差額例

X1turboIIIにするなら
(CZ-870C＋CZ-870D)

下取り機種	差額
CZ-812C＋専用モニター	¥159,000
CZ-804C＋専用モニター	¥166,000
CZ-811C＋専用モニター	¥166,000
CZ-802C＋専用モニター	¥169,000
CZ-803C＋専用モニター	¥167,000
CZ-801C＋専用モニター	¥168,000
CZ-800C＋専用モニター	¥169,000

X1turboZにするなら
(CZ-880C＋CZ-600D)

下取り機種	差額
CZ-856C＋CZ-855D	¥202,000
CZ-852C＋CZ-850D	¥212,000
CZ-851C＋専用モニター	¥217,000
CZ-811C＋専用モニター	¥242,000
CZ-803C＋専用モニター	¥243,000
CZ-801C＋専用モニター	¥244,000

〒101-91 東京都千代田区神田郵便局私書箱135号
㊀AM10時〜PM7時 ㊡毎木曜

5号店	☎03-251-0531
7号店	☎03-253-4199
ニューセンター店	☎03-251-0987

(7月は休まず営業いたします。)

X1・MZシリーズ周辺機器他、ビッグ超特価の品揃え!

ボーナス特集、下取りセール実施中!詳細はお問い合わせか本誌7月号の広告をご覧ください。

本誌発売時には、下記価格表より、さらにお求めやすい価格に変更されている場合があります。

●富士通FM-TV152(カラーCRT TV 15インチ)‥¥89,800⇨¥59,800

本体

新発売!16ビットパソコン「MZ書院」

●シャープMZ-2861 大量入荷‥‥標準価格¥328,000
●シャープCZ-803C‥‥‥‥‥¥119,800⇨¥29,800
●シャープCZ-804C‥‥‥‥‥¥139,800⇨¥38,500
●シャープCZ-8020‥‥‥‥‥‥¥69,800⇨¥39,800
●シャープCZ-822C‥‥‥‥‥¥118,000⇨¥78,000
●シャープCZ-850C‥‥‥‥‥¥168,000⇨品 切
●シャープCZ-870C‥‥‥‥‥¥168,000⇨¥128,000
●シャープCZ-880C‥‥‥‥‥¥218,000⇨アイビット価格
●シャープCZX1 68000‥‥ ¥369,000⇨ 即納可 !
●シャープMZ-2200‥‥‥‥‥¥128,000⇨¥29,800
●シャープMZ-2520‥‥‥‥‥¥159,800⇨大特価
●シャープMZ-2521‥‥‥‥‥¥198,000⇨¥89,800
●シャープMZ-5521‥‥‥‥‥¥388,000⇨アイビット価格
●NEC PC-9801VF2‥‥‥‥‥¥348,000⇨¥180,000
●NEC PC-8801mkIIMR‥‥‥¥238,000⇨¥128,000
●NEC PC-88VA‥‥‥‥‥‥¥298,000⇨アイビット価格
●NEC PC-8801FH(30)‥‥‥¥168,000⇨¥134,000
●NEC PC-9801VX2‥‥‥‥¥433,000⇨¥346,000
●NEC PC-9801UV21‥‥‥‥¥390,000⇨アイビット価格
●富士通FM-77AV20-2‥‥‥¥168,000⇨¥95,000

拡張機器他

●シャープCZ-8EB-3(X1拡張I/Oボックス)‥‥‥¥28,000
●シャープCZ8EP(X1拡張ボード)・¥11,800⇨¥10,000
●シャープMZ-8BGK(80B用拡張 I/Oボード)‥¥39,000⇨¥22,000
●シャープMZ-1U01(2000用拡張)‥¥37,000⇨(在庫切れ)
●シャープMZ-1U02(3500用拡張)‥¥20,000⇨¥7,000
●シャープMZ-1U03(700用拡張)‥¥35,000⇨¥15,000
●シャープMZ-1U05(5500用拡張)‥¥12,000⇨¥8,500
●シャープMZ-1U09(2500用拡張)‥‥¥9,000⇨¥7,200
●シャープMZ-8BK(80Bの拡張)‥¥19,800⇨¥12,000
●シャープ1R01+1R02×2‥‥‥¥55,000⇨¥18,000
●シャープMZ-2200用キーボード‥‥‥‥¥10,000
●シャープMZ-8BG‥‥‥‥‥‥¥39,000⇨¥19,800
●シャープMZ-1E24 232Cカード‥¥19,800⇨¥16,800
●シャープCZ8BR1(立体映像セット)‥¥29,800⇨¥25,300
●シャープCZ-8BK3(第2水準漢字ROM)‥¥13,800⇨¥11,800
●シャープCZ-8BK4(第2水準漢字ROM)‥‥‥¥6,800⇨¥5,700
●シャープMZ-1T02‥‥‥‥‥‥¥19,800⇨¥8,500
●シャープMZ-1M03(数値プロセッサー)‥¥69,000⇨¥35,000

●シャープCZ-8VC(RFビデオコンバーター)‥‥¥15,800⇨¥13,400
●シャープMZ-8BI04(GPIBカード)‥¥45,000⇨¥18,000
●シャープMZ-8BC04(GPIB ケーブル)‥‥¥18,000⇨¥8,500
●シャープMZ-1R09(5500用VRAM)‥‥‥¥35,000⇨¥25,000
●シャープMZ-1R10(5500用漢字ROM)‥‥¥30,000⇨¥12,000
●シャープMZ-1R11(5500用256RAM)‥‥¥80,000⇨¥40,000
●シャープMZ-1R14(5500用辞書ROM)‥‥¥40,000⇨¥20,000
●シャープMZ-1R18(1500RAMファイル)‥‥¥18,000⇨¥12,000
●シャープMZ-1R19(5500用第二漢字ROM)‥‥¥35,000⇨¥15,000
●シャープMZ-1R23(漢字ROM)‥‥¥19,800⇨¥12,000
●シャープMZ-1R24(辞書ROM)‥‥¥22,000⇨¥12,000
●シャープMZ-1R26(増設RAMボード)‥‥¥35,000⇨¥13,000
●シャープMZ-1R27(増設ビデオRAM)‥‥¥20,000⇨¥11,000
●シャープMZ-1R28(MZ-2500辞書ROM)‥‥¥22,000⇨¥13,000
●シャープMZ-1R29(1P17第2水準ROM)‥‥¥32,000⇨¥15,000
●シャープMZ-1R37(MZ-2500RAMファイル)‥‥¥35,800⇨¥29,800
●シャープMZ-1T03データレコーダー‥¥12,000⇨¥8,500
●シャープCZ-8BGR2(X1ターボ用)‥¥14,800⇨¥4,000
●シャープ CZ-8BS1(ステレオFM音源ボード)‥‥‥¥19,500
●NEC PC9808数値プロセッサー ¥82,000⇨¥30,000
●シャープCZ-6PV1(ビデオプリンター)‥¥198,000⇨新発売!
●シャープCZ-51F(X1ターボ増設)‥¥39,800⇨¥25,000
代 品
●シャープCZ-52F(X1F増設)‥‥‥¥34,800⇨¥22,000
●シャープMZ-2000/2200/80B/700用(フロッピーインターフェースカード)
‥‥‥‥‥‥‥‥‥‥‥‥‥‥‥‥‥¥18,000
●シャープMZ-6P11(1P11カットシート)‥¥95,000⇨¥35,000
●シャープMZ-1E15(1.2MミニFDインターフェイス)‥¥35,000⇨¥28,000
●シャープMZ-1E05(2000/80B/1500/700ミニFDインターフェイス)
‥‥‥‥‥‥‥‥‥‥‥‥¥24,000⇨¥18,000

プリンター

●シャープMZ-1P17(カラー漢字プリンター)¥79,800⇨¥39,800
●シャープMZ-1P09(MZ-1500用ケーブル付)‥‥¥47,600⇨¥15,000
●シャープCZ-8PP2(X1・MZ使用可)‥‥¥54,800⇨¥9,800
●シャープCZ-8PK2(漢字)‥‥‥¥134,800⇨¥39,800
●シャープCZ-8PD3‥‥‥‥‥‥‥¥59,800⇨¥29,500
●シャープMZ-1P10(漢字)‥‥‥‥¥245,000⇨¥95,000
●シャープ CZ-8PC1(熱転写カラープリンター)‥‥‥¥53,800
●シャープCZ-8PD2(ドットプリンター)‥‥‥‥‥‥¥29,500
●NEC PC-PR405-01(2水準漢字)・¥23,800⇨¥11,500
●NEC-PR101L(24ドット漢字プリンタ)‥‥‥¥175,000⇨¥59,800
●日立MP-1053(漢字プリンター16インチ)‥¥315,000⇨¥158,000

●シャープCZ-8PK3(24ドットプリンタ)‥‥¥189,000⇨¥129,800
●シャープCZ-8PK4‥‥‥‥‥‥¥158,000⇨¥99,800

フロッピーディスク

●シャープCZ-503F(5″2D×1)(インターフェースケーブル付)‥‥¥42,000
●シャープCZ-502F(5″2D×2)(インターフェースケーブル付)‥‥¥75,500
●シャープMZ-1F07(インターフェースケーブル付)¥158,000⇨¥95,000
●ラウンドシステムLDS-5UV(UV2ディスク)
‥‥‥‥‥‥‥‥‥‥‥‥¥78,000⇨¥65,000

ソフト

●シャープMZ-2Z013(5500MSDOS)¥25,000⇨¥21,000
●シャープMZ-2Z017(5500BASIC3)¥20,000⇨¥17,000
●シャープMZ-2Z023(5500/GWBASIC)‥¥50,000⇨¥42,500
●シャープMZ-2Z032(1500DIKBASIC)‥‥¥12,000⇨¥6,000
●シャープMZ-8BD02(80BF、DOS)¥50,000⇨¥15,000
●シャープMZ-2000 CP/Mデジタルリサーチ‥¥35,000
●シャープMZ-80B CP/Mデジタルリサーチ‥‥¥35,000
●シャープMZ-2Z004(2000/F.DOS)‥‥‥¥50,000⇨¥42,500
●シャープMZ-2Z005(2000/システムプログラム)‥¥25,000⇨¥21,500
●シャープMZ-1Z010(2000/232CGP.1B)‥‥¥9,500⇨¥8,500

16ビットボードキット

●MZ-1M01+漢字ROM‥‥‥‥‥‥‥‥‥‥¥18,000

※掲載されている商品は全て新品保証付きです。

全国通信販売

北海道から沖縄まで

信用をモットーに、よりよい品をより安く、迅速にお届けします。

★送料はご注文の際にお問い合わせ下さい。
★掲載の商品は、すべて新品、保証書付きです。
★掲載の商品は充分用意してありますが、ご注文の際は、在庫の確認の上、現金書留または、銀行振込でお申し込み下さい。全商品クレジットでも扱っております。
★お申し込みの際は必ず電話番号を明記して下さい。
★商品、品切れの節はご容赦下さい。

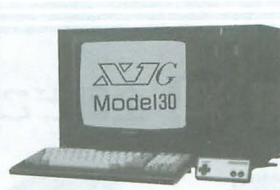
アイビット電子㈱

営業所:〒192東京都八王子市北野町560-5
☎0426-45-3001〜3
☎03-545-0022 FAX.0426-44-6002
●営業時間:10:00〜19:00
●電話受付:20:00迄可
●定休日:日曜日(祭日営業)

超優良中古パソコンが電話一本で買える!!

☎03(797)1221

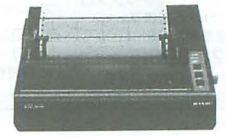
CZ-811CE(X-1Fモデル10)
¥89,800➡ ***¥26,800*** 新品
X-1Fモデル10ディスプレイセット
(本体+CU14GE)
¥139,600➡ ***¥56,600***

CZ-820CB(X-1Gモデル10)
¥69,800➡ ***¥39,800*** 新品
¥69,800➡ ***¥34,800***
X-1Gモデル10RF コンバータセット(本体+AN-58C)
¥72,780➡ ***¥37,600***
X-1Gモデル10ディスプレイセット
(本体+CU14GB)
¥119,600➡ ***¥64,600***

CZ-822CB(X-1Gモデル30)
¥118,000➡ ***¥78,000*** 新品
CZ-820DB
(14インチ2000字RGBTV)
¥79,800➡ ***¥44,800*** 新品同様
X-1Gモデル30ディスプレイセット
(本体+CU14GB)
¥167,800➡ ***¥107,800***

MZ-1P17(E・B)
(色、グレー・ブラック)
(80桁カラー漢字サーマルプリンタ)
¥76,600➡ ***¥42,800*** 新品
(X1用ケーブル付)
¥76,600➡ ***¥46,800*** 新品
(MZ2500用ケーブル付)

CU-14G(E・B)
(色、グレー・ブラック)
(14インチ2000字デジタルカラー)
¥49,800➡ ***¥29,800*** 新品同様

CZ-820DB
(14インチ2000字RGBTV)
¥79,800➡ ***¥44,800***

CU-14A4
(14インチ4050字アナログ・デジタルカラー、PC用アナログケーブル付)
¥89,800➡ ***¥59,800*** 新品同様

CZ-8PK2
(10インチ9ドット漢字プリンタ、X1用プリンターケーブル標準添付)
¥134,800➡ ***¥39,800*** 新品

SHARP

本体

商品	定価	価格
MZ721(データレコーダ内蔵)	¥89,800	¥12,000
MZ731(データレコーダ・カラープロッタ内蔵)	¥128,000	¥18,000
MZ-1500(高速クイックディスク内蔵、RF出力付き)	¥89,800	¥25,000
MZ-2000(GRAM、1、2、3ページ内蔵)	¥265,000	¥32,000
MZ-2200+MZ1T02(本体+専用データレコーダ付き)	¥147,800	¥24,500
MZ-6541(16ビット、5インチFD×2)	¥650,000	¥98,000
CZ-800C(X-1マニアタイプ、Gラム付)	¥187,000	¥20,000
CZ-801C(X-1C)	¥119,800	¥20,000
CZ-803C(X-1Cs)	¥119,800	¥20,000
CZ-804C(X-1Ck)	¥139,000	¥22,000
CZ-811C(X-1Fモデル10)	¥89,800	¥22,000
CZ-802C(X-1D)	¥198,000	¥22,000

プリンタ

商品	定価	価格
CZ-81P(80桁カラープロッタプリンタ)	¥34,800	¥12,000
CZ-8PP2(カラープロッタプリンタ)	¥54,800	¥12,000
MZ-1P01(MZ-700用カラープロッタ、アダプター付き)	¥39,800	¥8,000
MZ-1P14(80桁9ドットプリンタ)	¥54,800	¥20,000
MZ-1P09(カラープロッタプリンタ)	¥39,800	¥15,000

その他

商品	定価	価格
MZ-1F11(MZ700用QDドライブ)	¥24,800	¥10,000
MZ-1S01(MZ-1D02用チルトスタンド)	¥12,000	¥3,800
MZ-1S08(MZ-1D06用チルトスタンド)	¥12,000	¥3,800
CZ-300F(X-1用3インチフロッピーディスク)	¥79,800	¥15,000
CE-124(カセットインターフェース)	¥4,500	¥2,000

商品	定価	価格
CE-152(ポケコン用カセットレコーダ) 新品同様	¥19,800	¥7,500
CE-157(PC-1500用カナモジュール)	¥25,000	¥5,800
MZ-1M10(MZ-2500用カラーパレットボード)	¥14,500	¥8,000
MZ-1R26(MZ-2500用増設RAMボード)	¥35,000	¥15,000
MZ-1V09(MZ-2500用拡張ユニット)	¥9,000	¥5,800
CU-20P1(20インチ4050字デジタルカラー)	¥218,000	¥55,000
CZ-850DR(14インチ4050字RGBTV) 新品同様	¥129,800	¥59,800
X-1シリーズ特選極上品コーナー		
X-1Fモデル10(CZ-811CE、高速電磁カセットレコーダ内蔵) 新品同様	¥89,800	¥26,800
X-1Gモデル10(CZ-820CB、高速電磁カセットレコーダ内蔵) 新品同様	¥69,800	¥39,800
X-1Gモデル30(CZ-822CE、5″2D・FDD×2、漢字ロム付)	¥118,000	¥78,000
ディスプレイ特選極上品コーナー		
MD-12P1(12インチ4050字グリーン) 新品同様	¥39,800	¥29,800
CU-14G(14インチ2000字デジタルカラー) 新品同様	¥49,800	¥29,800
CU-14A4(14インチ4050字アナログデジタルカラー) 新品同様	¥89,800	¥59,800
その他特選極上品コーナー		
CZ-8PK2(10インチ9ドット漢字プリンタ) 新品	¥134,800	¥39,800
CZ-8PP2(S)(カラープロッタプリンタ) 新品同様	¥54,800	¥15,000
CZ-8VC(X-1用RFビデオコンバータ) 新品	¥15,800	¥13,800
MZ-1P09(MZ-1500用カラープロッタプリンタ) 新品同様	¥47,600	¥25,000
MZ-1P17(E・B)(80桁カラー漢字サーマルプリンタ X1用ケーブル付) 新品	¥76,600	¥42,800
MZ-1P17(E・B)(80桁カラー漢字サーマルプリンタ MZ2500用ケーブル付) 新品	¥76,600	¥46,800

C.B.サポートホットライン
☎03(797)1234

当社でコンピュータをお買い上げいただいたお客様に万一、トラブルが発生した場合、このホットラインで親切に対応いたします。

C.B.レスキューシステム

お客様のお手元でトラブルが発生した場合、当社より引取りにお伺い致します。万一、お買いになった機械が故障しても安心です。

⇧掲載の商品はいずれも限定品ですので今すぐお電話下さい。

★電話1本で高額買取り、即現金お支払い!★

- コンピュータバンクではあなたの不要になったパソコンを電話1本で査定し買取ります。
- どんな問い合わせにも親切に対応いたします。

▼本社注文デスク

☎03(797)1221

コンピュータバンク

株式会社パシフィックコンピュータバンク

〒150 東京都渋谷区渋谷1-6-8 井上ビル
営業時間/AM9:30〜PM10:00 年中無休

全商品保証付 6ヶ月の保証期間だから安心です。

全国無料配送 全国どこでも配達料はいただきません。

高額下取り 少ない予算で買いかえもラクラク。

代金引換えシステム 商品到着時の代金支払いでOK。

クレジットでOK カレッジクレジットも取扱います。

日曜配達可 留守の多い方でも安心です。

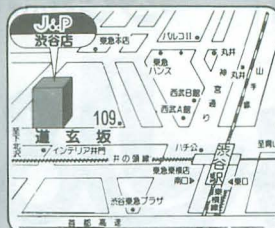
高額買取り 電話1本で即、現金お支払い。

ボーナス一括払い 商品は即お手元へ、お支払いはボーナス時に。

ピング

メールショッピングのお申し込みは J&P 渋谷店で承ります。

フロアーごあんない

階	売場
4F	パソコン教室 ●パソコン入門コース●BASIC上級コース ●BASIC初級コース●各種ビジネスコース
3F	OA機器 ●ビジネスパソコン●ワードプロセッサ ●ビジネスソフト●OAサプライ ●ハンドヘルドコンピュータ
2F	ビジネスパソコン ●パソコン●ディスプレイ ●プリンタ●専門書籍 ●パソコンアクセサリー
1F	ホビーのパソコン ●ホビーパソコン●MSX ●ゲームソフト●学習ソフト

東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141

M8-333 **■ディスク価格表** (いすれも10枚単位になっております。)

	5"2D	5"2DD	5"2HD	3.5"1DD	3.5"2D	3.5"2DD	3.5"2HD
マクセル	❶¥2,800	❽¥3,800	⓯¥5,000	㉒¥6,300	㉙¥6,300	㊱¥7,800	㊸¥13,500
スリーM	❷¥2,800	❾¥3,800	⓰¥5,000	㉓¥6,300	㉚¥6,300	㊲¥7,800	㊹¥13,000
メヨレックス	❸¥2,800	❿¥3,800	⓱¥5,000	㉔¥6,300	㉛¥6,300	㊳¥7,600	㊺¥13,100
データライフ	❹¥2,800	⓫¥3,200	⓲¥4,900	㉕¥5,500	㉜¥5,500	㊴¥6,100	㊻¥12,200
フジ	❺¥3,200	⓬¥4,600	⓳¥6,300	㉖¥6,700	㉝¥6,700	㊵¥8,400	㊼¥14,500
ソニー	❻¥3,200	⓭¥4,600	⓴¥6,000	㉗¥7,100	㉞¥7,100	㊶¥8,900	㊽¥14,600
TDK	❼¥2,600	⓮¥3,600	㉑¥4,700	㉘¥6,100	㉟¥6,100	㊷¥7,500	㊾¥12,500

㊿J&Pオリジナルディスク
MD-2HD 10枚
¥3,800

■〈MZ-2500オプション〉

M8-334

MZ-1E26
J&P価格**24,800円**
ボイスコミュニケーションインターフェイス

M8-335

MZ-1M10
J&P価格**14,500円**
カラーパレットボード

M8-336

J&P価格**10,000円**
MZ-1M08
MZ-2500/1500用
ボイスボード

M8-337

MZ-1R28
J&P価格**22,000円**
MZ2500用、辞書ROM

■ポケコン アクセサリー

M8-338

①CE-124 J&P特価**4,000円**
PC-1245〜1360用 カセット インターフェイス
②CE-202M J&P特価**16,000円**
PC-1350·1360·1450·7500用 16KBメモリ
③CE-2H32M J&P特価**28,000円**
PC-1360·1360K·1460用 32KBメモリ
④CE-2H16M J&P特価**14,000円**
PC-1360·1360K·1460用 16KBメモリ

■〈X-1/ターボオプション〉

M8-339

FM音源ボード
シャープCZ-8BS1 J&P価格**23,800円**
X-1用8重和音200音色、ステレオ サウンドのFM音源

M8-340

立体映像セット
シャープCZ-8BR1
J&P価格**29,800円**
X-1/X-1ターボシリーズにて立体映像が楽しめます。
立体作画ソフト·立体スコープ付

M8-341 M8-342

マウス
シャープCZ-8NM1
J&P価格**13,800円**
X-1·MZ用マウス

カラーイメージボード
シャープCZ-8BV2
J&P価格**39,800円**
画像を自在に修正·加工できます
画像処理ツール·グラフィックソフト同梱

■プリンタオプション M8-343

❶MZ-1C48 X-1用プリンタケーブル	6,800円
❷MZ-1C35 MZ-2500/2200/2000用ケーブル	6,800円
❸MZ-1R29 MZ-1P17(B)用第2水準ROM	14,800円
❹CZ-8PC1-3 CZ-8PC1用第2水準ROM	9,800円

■MZ-2500システムソフト M8-344

商品名	機種名	価格
FORTRAN	❶IP-1213	13,800円
C言語	❷IP-1214	13,800円
COBOL	❸IP-1215	13,800円
LISP	❹IP-1216	13,800円
PROLOG	❺IP-1217	13,800円
CPM	❻MZ-6Z001	16,800円

■X-1/X-1ターボ システムソフト M8-345

商品名		機種名	価格
ランゲージマスター(CP/M®)		❶CZ-128SF(2D·5"FD版)	9,800円
turbo CP/M(漢字版)		❷CZ-130SF(2D·5"FD版)	14,800円
ミュートピア		❸CZ-139SF(2D·5"FD版)	12,800円
ランゲージシリーズ	FORTRAN	❹CZ-115LF(2D·5"FD版)	13,800円
	C	❺CZ-116LF(2D·5"FD版)	13,800円
	turbo LOGO(漢字版)	❻CZ-117SF(2D·5"FD版)	18,800円
	COBOL	❼CZ-118LF(2D·5"FD版)	13,800円
	PROLOG	❽CZ-119LF(2D·5"FD版)	13,800円
	LISP	❾CZ-120LF(2D·5"FD版)	13,800円
	APL	❿CZ-126LF	13,800円

■X-1をパワーアップさせるNEW BASIC (Ver.2.0)

M8-346

対応機種	NEW BASIC	価格
CZ-800C CZ-801C CZ-802C CZ-803C CZ-804C	❶カセット版CZ-112SF	7,800円
	❷3"FD版 CZ-113SF	8,800円
	❸5"FD版 CZ-124SF	8,800円

■各種漢字ROM M8-347

❶CZ-8BK2 X-1F第1水準ROM	19,800円
❷CZ-8BK3 X-1ターボ第2水準ROM	13,800円
❸CZ-8BK4 X-1ターボ2第2水準ROM	6,800円

お申し込み方法

右の注文書にご希望商品の注文No.および必要事項ご記入の上、現金書留にてJ&P渋谷店までお申し込みください。現金受領後、発送いたします。
なお、現金書留以外で申し込まれた場合は責任を負いかねます。

●記載以外のご注文も承りますので、詳しくはお電話にてお問い合わせ下さい。
☎(03)496-4141

キリトリ線

現金書留申込み用紙

おところ 〒□□□□□

TEL ( )

おなまえ 様

注文No.	数量	金額
M8- ( )		円
M8- ( )		円
合計		円

お手持ちのパソコン

お申込み先：東京都渋谷区道玄坂2丁目28番4号(〒150) J&P 渋谷店メールショッピング係

ピング

メールショッピングのお申し込みは J&P 渋谷店で承ります。

東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141

■X-1シリーズ5インチディスク版

X1ユーザー話題のワールドマップシュミレーションゲーム。国外に持ち出されたICカードを、世界169ヵ国の名所、旧跡を訪ねての調査、追跡。事件解決のあかつきには、キミは世界地理の"通(オーソリティー)"。

タイトル	ワールドイングス169
適応機種	M8-71 X1ターボZ M8-72 ／X1ターボシリーズ
価格	X1ターボZ用(5"2HD)12,500円／X1ターボシリーズ用(5"2D)11,000円
ソフトハウス	アイ・ヴィ・アイ

タイトル	ラスベカス	棋太平(対局将棋)	グーニーズ	殺人倶楽部	森田和郎の将棋	迷宮の扉	ザナドウ	ドラゴンバスター
適応機種	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T
ソフトハウス	スタークラフト	S・P・S	コナミ	リバーヒルソフト	エニックス	デンパ	日本ファルコム	デンパ
注文No 価格	M8-73 ¥7,800	M8-74 ¥6,500	M8-75 ¥6,800	M8-76 ¥7,800	M8-77 ¥7,800(5"2D)	M8-78 ¥6,200	M8-79 ¥7,800	M8-80 ¥6,200
タイトル	プロフェッショナル麻雀	グラディウス	アルバトロス	ファイナルゾーン	スーパーマリオブラザーズSP	ザナドゥ・シナリオII	夢幻戦士ヴァリス	大戦略X1
適応機種	X-1T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T
ソフトハウス	シャノアール	コナミ	日本テレネット	日本テレネット	ハドソン	日本ファルコム	日本テレネット	システムソフト
注文No 価格	M8-81 ¥6,800	M8-82 ¥6,800(5"2D)	M8-83 ¥8,800	M8-84 ¥6,800	M8-85 ¥6,800	M8-86 ¥5,800	M8-87 ¥7,800	M8-88 ¥6,800
タイトル	蒼き狼と白き牝鹿	九五伝	ハイドライドII	ロマンシア	覇邪の封印	トップル・ジップ	めぞん一刻	カーマイン×1
適応機種	X-1T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T
ソフトハウス	光栄	テクノソフト	T&Eソフト	日本ファルコム	アスキー	ボーステック	マイクロキャビン	マイクロキャビン
注文No 価格	M8-89 ¥7,800	M8-90 ¥7,800	M8-91 ¥6,800	M8-92 ¥6,800	M8-93 ¥8,800	M8-94 ¥6,800	M8-95 ¥6,800	M8-96 ¥7,800
タイトル	ウィバーン	ウィザードリー2	エルスリード	アルゴー	信長の野望(全国版)	うっでい・ぽこ	三国志	未来
適応機種	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T	X-1/F/T
ソフトハウス	アルシスソフト	アスキー	NCS	呉ソフト	光栄	d Bソフト	光栄	ザインソフト
注文No 価格	M8-97 ¥6,800	M8-98 ¥9,800	M8-99 ¥7,200	M8-100 ¥7,800	M8-101 ¥9,800	M8-102 ¥6,800	M8-103 ¥14,800	M8-104 ¥7,800

お奨めソフト

ご定評をいただいている〈即戦力〉が高度な機能・操作性にさらに磨きをかけ、お求めやすい価格で新登場です。

M8-116 高性能日本語ワープロ
即戦力Samurai(侍)
X1/X1 tubo用5"2D
¥19,800(サムシンググッド)

注文No	適応機種	タイトル	ソフトハウス	メディア	価格	価格
M8-105	MZ-2500	ユーカラK2	東海クリエイト	3.5"DD	¥28,000	一括入力、逐次文節変換方式の日本語ワープロ、文節学習機能も装備。ブロック入力をはじめとした強力な編集機能も特長。
M8-106	X-1ターボ	ビジレス漢字版	OAテック	5"2D	¥48,000	カンタン操作で自由な表づくり。項目別検索。セル間演算。集計。自動プログラムと機能も充実。
M8-107	X-1ターボ	日本語ワープロ「即戦力」	サムシンググッド	5"2D	¥39,800	99%の変換達成率を可能にした使いやすさ。16ビットに迫る機能を実現!
M8-108	X-1ターボ	Multiplan	シャープ	5"2D	¥49,800	16ビット機でしかなかったあのマルチプランがX-1ターボで新発売、ビジネスにはぜひ活用したいソフトです。
M8-109	X-1ターボ	ユーカラPOP	東海クリエイト	5"2D	¥28,000	ワープロと通信ソフトがドッキング、各種B・B・S局への通信やデータベースへの交信に使用できます。
M8-110	X-1ターボ	日本語 My CARD	アバロン	5"2D	¥58,000	マイコン表示による使い易さと独自のOSによる超高速処理のカード型データベース。
M8-111	X-1ターボ	Z'sSTAFF	シャープ	5"2D	¥19,800	X1ターボシリーズの優れたグラフィック機能を存分に発揮させる待望の本格グラフィックツールです。
M8-112	MZ-2500	TURBO PASCAL (Ver3.0)	MSK	3.5"2DD	¥29,000	最強・低価格のPascalコンパイラーがMZ-2500でもご利用いただけます。
M8-113	X-1ターボ	Inkpot(マウス付)	アスキー	5"2D	¥38,000	エアブラシを含む14種類のペン先と37種類のタイトルパターンを用意しました。マウスを使って、多彩な編集機能で映像をコントロール。
M8-114	X-1ターボ	印刷工房	モーリン	5"2D	¥14,000	24ドットプリンタ以外でも24ドット印字を可能にします。1／4角、網かけ、斜体、強調印字もでき文書表現も豊かにします。(ユーカラが必要)
M8-115	MZ-2500	カラー印刷キットぱれっと	ダイナウェア	3.5"2DD	¥18,000	「ぱれっと」は絵や文字を組み合せた表現豊かなカラーグラフィックを手軽に描いて印刷できるソフトです。(マウス別売)

お申し込み方法

右の注文書にご希望商品の注文Noおよび必要事項ご記入の上、現金書留にて J&P 渋谷店までお申し込みください。現金受領後、発送いたします。
なお、現金書留以外で申し込まれた場合は責任を負いかねます。

●記載以外のソフトのご注文も承りますので、詳しくはお電話にてお問い合わせ下さい。 ☎(03)496－4141

キリトリ線

現金書留申込み用紙

おところ 〒□□□□□

TEL （ ）

おなまえ 様

注文No(ディスクサイズ)	数量	金額
M8- （ ）	本	円
M8- （ ）	本	円
M8- （ ）	本	円
合計	本	円
お手持の機種名 （ ）		

お申込み先：東京都渋谷区道玄坂2丁目28番4号(〒150) J&P 渋谷店メールショッピング係

X-68000専用

20M・HDDはあなたのワークステーションを存分にパワーアップ!

グラフィックス・アニメは、フロッピーディスクに2面しか入りません。また、すべての標準ソフトをビジュアルシェルできませんでしたが、この20Mハードディスクがあれば、グラフィックスや、ビジネスデータを大量に保存できます。使い方次第で、存分にパワーアップできます。

● アニメは、フロッピーディスクに2面しか入りません。

はじめての人でもセッティング可能なアプリケーションソフトを同梱しました。

どなたにもHDDが簡単にセットできる様にしました。安心してご使用いただけます。
専用インターフェイスケーブルもついてます。

X-68000本体と同色。

X68000の人気の秘密に、すぐれたデザインもあります。本HDDは、その美的感覚をそこなうことなく、システムアップします。

格安で高性能。

20Mで、プリンター程度の価格におさえました。
X68000のユーザーのシステムアップに、心よりお手伝いします。

お申し込み方法 —全国通信販売—

ご注文の際は、在庫の確認の上、現金書留または銀行振込でお申し込み下さい。
送料は、ご注文の際にお問い合わせ下さい。
商品は、すべて新品、保証書付きです。

商品内容

● I/Fケーブル ● アプリケーション HDD MAKE ソフト ● 保証書

P,S,H あなたはX-68000をワークステーションとして生かしていますか。/発売記念価格158,000円(定価198,000円)

20M(メガ) HDD(ハードディスク)新登場

●お問い合わせ、お申し込み/パソコンショップハドソン 札幌市中央区南1条西2丁目[illegible]井今井3F ☎(011)241-5367

SHARP

Oh!MZ
8月号 昭和62年8月1日発行（毎月1回1日発行）通巻第64号
昭和58年11月2日第三種郵便物認可

アナログカラーイメージボード内蔵

ビデオやテレビなどの映像を最大4,096色のリアルさで瞬時に取り込み表示。モザイク処理や反転、階調を変える量子化処理など多彩な取り込み機能をサポートしたグラフィックツールも同梱、アイコン表示とマウス入力で手軽に画像処理やC.G.作成が楽しめます。表示能力も200ライン4,096色同時表示、400ライン4,096色中8色表示とパワーアップされています。

4,096色対応ニューテロッパ機能

4,096色のコンピュータ画像はもちろん、テレビやビデオ映像などと重ね合わせたスーパーインポーズ画像もビデオに録画でき、オリジナルビデオづくりが楽しめます。

8重和音ステレオFM音源搭載

L・R2チャンネルのオーディオ出力によりダイナミックなステレオシンセサイザーサウンドの世界が拡がります。200音色を標準で装備したミュージックツールも同梱。

マウス標準装備

クリエイティブワークがフレンドリーに、複雑な作画入力も簡単操作で楽しめます。

JIS第1/第2水準漢字ROM実装

難しい人名や地名もスピーディに表示、住所録や名簿も美しく仕上がります。

システム・ユーザー辞書装備

音訓・部首索引で検索できる第2水準漢字をサポート。専用辞書としても使えます。

1Mバイト5インチフロッピー2基搭載

大容量ファイルとしてはもちろん、従来の豊富なソフトも活かせる設計です。

X1ターボが誇るパフォーマンスを継承

高度な能力で定評の漢字BASIC/多彩な通信ツールのサポートで手軽なパソコン通信。

シャープ株式会社 ●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
電子機器事業本部テレビ事業部第4商品企画部〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

資料請求券 X1 turbo Z oh!MZ 8月

雑誌 02179-8 T4910217908487 ㈱日本ソフトバンク発行 Printed in Japan 定価480円 Oh!MZ